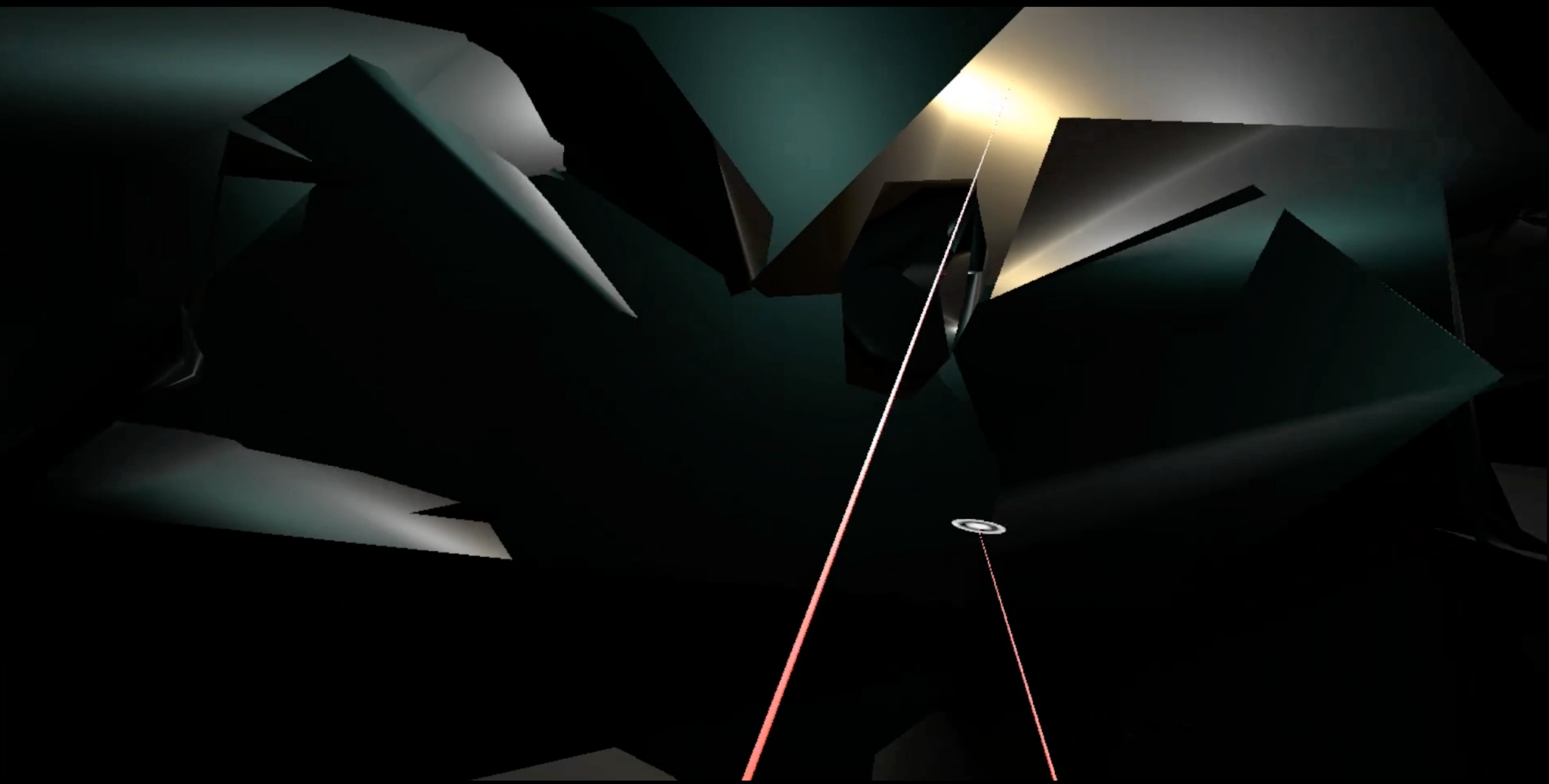
J. Camilo Sánchez Carranco

VIBRATING INSTRUMENTS IN VIRTUAL REALITY

A cohesive approach to the design of Virtual Reality Musical Instruments



Vibrating Instruments In Virtual Reality: A cohesive approach to the design of Virtual Reality Musical Instruments

J. Camilo Sánchez Carranco

School of Arts, Design and Architecture

Espoo 10.11.2018

Supervisor and advisor

Koray Tahiroğlu

Aalto University
School of Art and Design

Aalto University
School of Art and Design

Aalto University, P.O. BOX 11000, 00076 AALTO
www.aalto.fi

Author J. Camilo Sánchez Carranco

Title Vibrating Instruments In Virtual Reality: A cohesive approach to the design of Virtual Reality Musical Instruments

Degree programme New Media

Major Sound in New Media **Code of major** ARTS3052

Supervisor and advisor Koray Tahiroğlu

Date 10.11.2018 **Number of pages** 48+9 **Language** English

Abstract

This thesis presents the design, implementation and findings of a Virtual Reality Musical Instrument (VRMI). The project was done under the direction of the Sound and Physical Interaction (SOPI) research group. The project was made following an iterative design methodology and the metaphors and design patterns used in Ubiquitous Music Systems.

In contrast with the fast adoption of Virtual Reality as a platform for new entertainment productions, it is noticeable that the area of new interfaces for musical expression (NIME) has been disbelieving towards this technology. At the same time, previous projects under the category of VRMI have made a clear distinction between the instrument, an external 3D model, and the user. Thereby, this thesis presents a project that focuses on how VR can enhance individual musical interaction? In order to do so, this project is directed to blurry the lines between performer, instrument and environment by creating immersion through 3D audio, audiovisual feedback, bodily and spatial interaction, the performer and the system's autonomous responses. As a final result, this thesis reaches to provide the NIME community with a purposeful use of Virtual Reality as an interactive musical platform.

Keywords NIME, Virtual Reality, Musical Instrument, Interaction Design, HCI,

Acknowledgments

I want to strongly thank Koray Tahiroğlu for his support and guidance not only during this project but through all my education in Media Lab. Your vision of music and technology has been extremely inspiring. Deep thanks to Antti Ikonen for his support, tutoring and trust during this years, my sonic perception evolved greatly under your teaching.

I want to thank Media Lab's Support Hub and IT staff, Ilpo, Pekka, Janne, Janne Ojala, Juuso, Elias and Joel. Thanks for all the good moments and conversations we have had while wrapping cables.

Huge thanks to all medialabbers and especially to those involved in ÄÄNIAALTO, Pikkuääniaalto and DASH'16: New Media Festival. Special gratitude to DADA: Media Lab's Student Association for all the effort and events they have done to support our community. I also want to thank the Music Technology Department of Sibelius Academy. Marianne, Andrew and Josue thanks for your teaching and knowledge.

Quiero dedicar un especial agradecimiento a Alejandro Olarte: inf.do{gracias} por todas esas enseñanzas, conceptos y formas de trabajar que me han influido profundamente tanto en éste como en otros proyectos.

Por supuesto, muchísimas gracias y amor a mi madre Marisol y a mi padre Álvaro por todo el apoyo que he recibido desde que vine a Finlandia.

This thesis would not have been possible without the help of Simon Brown, Bryant Hoban, Sofia Kolovskaya and all the participants of the user tests. Simon and Bryant, thanks a lot for proofreading this thesis. I really appreciate the effort you put into this and I was really happy with the final outcome. Sofia, many thanks for making my diagrams appealing, understandable and not a mess of arrows.

Johanna, Taimu and Chichi you became a family for me and I will always keep you in my heart.

Last but not least, Alex, Marloes, Matti and Victor much love for you guys. I have spent my best moments in Finland with you and you are not only a source of happiness but also of knowledge and inspiration. Thanks.

Helsinki, 02.10.2018

J. Camilo Sánchez Carranco

Contents

Figures and tables

List of Figures

List of Tables

Abbreviations

DIY	Do-It-Yourself
FOV	Field Of View
HMD	Head Mounted Display
HRTF	Head-Related Transfer Function
VMI	Virtual Musical Instrument
VR	Virtual Reality
VRMI	Virtual Reality Musical Instrument

1 Introduction

This thesis is part of the *Vibrating Instruments in Virtual Reality* (VIVR) project of the Sound and Physical Interaction (SOPI) research group from the Department of Media at Aalto University. It presents the design and implementation of a homonymous VRMI completed by the author of this thesis, under the direction and guidelines of SOPI's head of research Koray Tahiroğlu.

SOPI's research focuses in Sound & Music and Computing & Sonic Interaction Design. Around these two fields, SOPI develops the concepts of real-world physical interactions in digital environments (Niinimäki & Tahiroglu, 2012), audio-tactile augmentation (Lai, Niinimäki, Tahiroglu, Kildal, & Ahmaniemi, 2011), embodied interaction (Tahiroğlu, Kildal, Ahmaniemi, Overstall, & Wikström, 2012), smart technologies (Tahiroğlu, Svedström, & Wikström, 2015; Tahiroğlu, Correia, & Espada, 2013) and new interfaces for musical expression (NIME) (J. Vasquez, Tahiroğlu, & Kildal, 2017; Tahiroglu, Svedström, & Wikström, 2015).

In VIVR research, SOPI aim is to study the needs and problems behind the key features in musical practices with VR; understanding the notion of VR determining its NIME design principles; and building a VIVR's toolkit reflecting the principles that were deployed in the role of emerging VR technologies in music practices. In particular, this thesis project addresses the question: *how an immersive Virtual Reality environment can enhance individual musical interaction?*. Hence, the design experience and practical implementation of a VRMI is used in order to present the NIME community with a purposeful use of VR as an interactive musical platform.

Since the beginning, VR has tried to create the perception of virtual experience being real. Sutherland's (1965, p. 508) initial statement was to make the virtual world look real, sound real and feel real. This eagerness of achieving high levels of immersion has brought the development of VR interfaces that allow up to six degrees of freedom, positional tracking, and high definition VR displays. Nonetheless, these technical developments are not enough to achieve immersion. In fact, it takes the ideation of a virtual experience to figure out how to use these technical developments in order to create a sense of immersion. Once this immersion is achieved, the user's perception is tricked into believing that what is happening in a virtual environment, it is actually happening in real life. This shift of perception can be used to bring a musical experience to a brand new level. It can not only make music visible, but it can change the approach to music making. As a matter of fact, today's technical limitations do not favour collective VR experiences. Hence, this thesis project focuses on the advantages that VR can provide when creating an individual music experience.

As for the concepts approached by SOPI, NIME seems to be the best classification for this thesis project. NIME projects span along different categories such as DIY digital musical instruments, augmentation of traditional instruments, software instruments, and the research about design, performance, and reception of them. At the same time, these projects are often an exploratory approach to music making in constant development (Morreale et al., 2018, p. 168). Partly because of this reason, NIME instruments/tools are not created with a mass audience in mind. Instead, they represent the artistic individuality and needs of a person or collective that has to develop their own tools in order to express their own artistic production (2018, p. 172). However, despite all the categories covered by NIME, only nine papers related to VR can be found in the NIME proceedings archives[1]. The limited amount of VR NIME articles implies the poor adaption of a technology that's features can be used in benefit of music interaction.

This thesis provides the findings of how VR's affordances can be implemented to develop individual music interaction. With this aim, the factors of 3D audio, audiovisual feedback, bodily and spatial interaction, autonomous responses and performer are merged as an ecosystem to favour the musical content. Some questions that are focused on along the different sections of this thesis are:

- How to shift the paradigm of playing a musical instrument when offered the possibility of creating new performance environments?
- How does 3D audio play an advanced role in music development?
- How to establish the same level of importance between sonic and visual reactions?
- How to reach playable interaction considering hardware limitations?
- Do the instrument's autonomous responses help to engage in music making?
- How to approach the previous questions to achieve immersion?

The following sections present an overview of the related work in the context of VR musical instruments. Secondly, the design and implementation of VIVR are introduced. This section describes the challenges and implementations[2] followed in order to answer the research question. Following, a set of user tests is presented. Next, a discussion on the user tests' results, the current design of the system and why the research of this topic should be supported are presented. Finally, a conclusion sums up the ideas and findings of this project.

[1] `http://www.nime.org/archives/`

[2] The thesis is supported by a series of code listings found in Appendix A. These listings are not the final scripts of the project but rather illustrative examples of the system's mechanics. For the full repository of the project, please refer to this link `https://bitbucket.org/krrnk/vivr/src/master/`.

2 Related work

The alteration of human visual perception is a technique used in western visual art since the Renaissance (Kubovy, 1988, p. 32). Nevertheless, the first idea of VR as we understand it today was presented by Ivan Sutherland (1965). In 1968, I. Sutherland invented what it is considered the first VR headset (LaValle, 2017, p. 30). Since then, VR technologies have evolved enough to become a commercial product of entertainment. From the sonic perspective, VR platforms have been used in combination with the latest developments in sound processing and audio spatialization. This connection has generated new systems that can facilitate musical interaction.

One of the most recent approaches to design principles for interactive music systems in VR comes from Serafin, Erkut, Kojs, et al. (2016). In this work, the authors make a distinction between *Virtual Musical Instruments (VMIs)* and *Virtual Reality Musical Instruments (VRMIs)*. According to Välimäki and Takala (1996, p. 10), the instruments of the first category are software simulations or extensions of existing musical instruments based on physical modelling synthesis techniques and combined with physical and/or gestural control interfaces. The VRMI category includes those musical instruments presented in a simulated virtual environment displayed on a Head Mounted Display or a multi-projection set-up (2016, p. 22). Taking into consideration these definitions, this thesis focuses on the second category.

Regarding VRMI's development, in 1987, Jaron Lanier introduced a VRMI for the piece *The Sound of One Hand*, a live improvisation in VR and performed with a just one hand in a DataGlove (Lanier, 1993). The hardware specifications of *The Sound of One Hand* show why it falls inside VRMI's denomination: a prototype XVR EyePhone from VPL (HMD), a DataGlove, and a magnetic tracker. In the design of *The Sound of One Hand's* instruments, Lanier already established some principles that have served as a base for the design of VRMIs. As seen in table 1, when displaying Lanier's design principles with the ones proposed by Serafin, Erkut, Nordahl, et al., one could argue that the former's are still relevant nowadays.

The following paragraphs present how these and other design principles have been used in the creation of VRMIs. When immersed in a 3D environment, one can expand the multiple forms of feedback from the performed instrument to create a multi-sensorial experience. This implementation is possible by providing a meaningful combination of audiovisual mappings together with haptic responses and bodily interaction. This combination of factors also extends the immersion of VRMIs (Serafin, Erkut, Kojs, et al., 2016, p. 26). The work of Mäki-Patola et al. (2005)

Lanier (n.d.)	Serafin, Erkut, Nordahl, et al. (2016)
Multi-sensorial experience	Audiovisual and haptic design in tandem
Follow conventions or shift paradigms	Consider both natural and magical interactions
Spatial mapping	Create a Sense of Presence
Simulation control	Represent the Player's Body
Automation	Do not copy but leverage expert techniques
Play in front of an audience	Make the experience social
	Reduce latency
	Consider the ergonomics of the display

Table 1: VRMI's design principles

presents an example of this audiovisual combination. Their Virtual Membrane is capable of visualizing the vibrations of a virtual plate in a 3D space, according to the plate dimensions, tension and velocity of user's strikes (2005, p. 14).

Simultaneously, since this multi-sensorial interaction happens in a virtual environment, it has the possibility of not following the natural laws of physics. As Lanier (1993), Serafin, Erkut, Kojs, et al. (2016, p. 28), and Hamilton and Platz (2016, p. 337) mention, one can use bodily and spatial interaction to change the parameters of the instruments, design inconceivable movements in the natural world, or use models of already existing ones but with a functionality that their natural counterparts' assembly would not allow. This has been seen in the case of *The Sound of One Hand* (Lanier, 1993) where the performer controlled three instruments with just one hand gesture, a limitation that allowed him to explore the instrument's control possibilities further.

Furthermore, Lanier (1993) adds that "simulation control", understood as the way the performer can hold and interact with the 3D model, is a critical design consideration of virtual hand tools when force-feedback is not available. Mäki-Patola's team (Mäki-Patola et al., 2005) presented an example that sets a compromise between this paradigm. Their *Air Guitar* is built upon conventional gestures used in guitar performance techniques, but simplified. In this manner, users can get a straightforward understanding of how it works, thus facilitating interaction with the instrument (2005, p. 15). Consequently, LaValle (LaValle, 2017, p. 221) suggests that a consideration on how physics are used in a 3D environment should be taken into account in order to not disrupt human learned concepts and have a sort of intuitive and pleasurable experience.

Even though interaction happens in a VR environment, this is still a defined and designed space. In relation to spatial mapping, Berkowitz et al. (2016, p. 341) discuss two approaches of using the virtual space as a component of the musical form. The “static" approach is where the virtual space does not have a role of an independent agent on the music, but it is the performer who coordinates with the musical interaction within that particular space. By contrast, the second approach proposes a “dynamic" space where autonomous changes and movement of objects inside the space affect music form and interaction. While the “static" approach provides a great exploration of the musical content and gives the musician a better control of shaping the form, it misses some of the possibilities that human-computer interaction can offer through this virtual space. For instance, these capabilities are generative music development, spatial interaction, artificial music gestures or improvement of sound interaction through machine learning among others (2016, p. 343). Alternatively, the “dynamic" approach provides the system with an agency role that can create a great variety of music and spatial interaction, but eliminates some the performer’s control (2016, p. 343). The discussion concludes that VR proves itself as a useful platform where works that follow any of these two approaches can be developed and recreated (2016, p. 344).

On the topic of spatial interaction, Serafin, Erkut, Kojs, et al. (2016, p. 28) consider the matter of the user’s presence. It is important to create an experience that allows the user to feel inside the VR environment. Users might react as if the whole experience is real if the designed illusion feels like they are living what is happening in this virtual space (Slater, 2009, p. 3554). In relation to VRMIs, these factors can be addressed by understanding the technical limitations of the device; emphasizing the audiovisual and haptic relations; reducing latency; creating a virtual representation of the user (2016, pp. 27-28); and considering elemental physic relations (LaValle, 2017, p. 221).

In order to expand development of music interaction in VR, Liang and Ming (1994) suggest the use of machine learning. Their research project is a VR system for music generation based on supervised learning. Following the hypothesis *“one’s behaviour is strongly related to the music style he prefers"*, the system would pick sounds or a progression of chords according to user’s movement based on a database of gestures related to different styles of music (1994, pp. 141-142). The result of their research shows that, even though the music performed with their system is interesting, it is oversimplified due to its constraints of just timbre and chord progression.

In accordance with Liang and Ming, Deacon et al. (2017) argue that machine learning should be a direction to follow when designing an interactive musical system in VR. Thus, machine learning can help improve design implications. According to their study, it can implement gestural natural behaviours by correcting interference caused by system hardware limitations. For example, this limitations can affect user experiences such as the FOV or user tracking. Furthermore, it can evaluate the human approach to learning a new system in order to provide different levels of interaction that can help users to engage deeper with the system (2017, p. 216).

Finally, addressing the socio-cultural element of playing music has turned out to be a challenge in VRMIs. As Serafin, Erkut, Kojs, et al. (2016, p. 29) point out, the current state of VR is still an individual experience. This is mostly due to the occlusive characteristics of HMDs, small adoption by the consumer market and limitations in networking technology. Likewise, Hamilton and Platz (2016, pp. 337-339) emphasize the disparity between performing music in VR and experiencing it as an audience. In their collaborative performance, some members of the ensemble wear an HMD and interact with the main instrument. Other musical components of the performance are pre-composed bell sequences and a laptop-orchestra ensemble that follows a director. In this case, the visualizations are presented to the audience on a front-facing screen. This divergence between the immersive three-dimensional views presented to each performer and the two-dimensional ones presented to the audience exposes one drawback of music performance in VR that still needs to be addressed.

The following sections present the methodology and design process of the VRMI created at SOPI. VIVR was developed taking into account the design principles of Serafin, Erkut, Nordahl, et al. (2016) but adding some of the ideas presented by other authors in this section. However, in contrast to the common approach of designing a 3D model that represents the instrument, VIVR approach to music making is based on an exploration of the virtual environment. Hence, the users do not perform an instrument that is external to their virtual representation, instead, they are placed inside the instrument. This design decision was made in order to enhance immersion in musical interaction.

3 Research material and methods

The original idea of the project was to design a VRMI with the aim of understanding if VR can provide new capabilities for individual music interaction. Taking into consideration that the works addressed in Section 2 were based on 3D models that represented an external instrument to the user, in SOPI, we wanted to follow a more exploratory approach. Hence, VIVR is designed as an instrument where the user is trapped in and that is actuated from inside. Different interaction and sound processing techniques were applied with this idea in mind and were changed according to discussions in the SOPI research group.

During the ideation stage of the project, a review suggested implementing the methodology used in Ubiquitous Music Systems. These systems are music environments based on a relationship between different ways of interaction and sound sources (Keller et al., 2014, p. XI). Some categories of Ubiquitous Music Systems are interactive installations, artistic interventions, eco-composition and cooperative composition. While being aware that VIVR is not a Ubiquitous Music System in all senses, the metaphors and design patterns used in the development of these systems were useful when implementing different ideas in VIVR. This VRMI does not fit all the characteristics of these systems since it is made for a specific device (HTC Vive); VR is not yet a widespread consumer technology; apart from the HTC Vive headset, it depends on a powerful workstation capable of running HTC Vive applications; the fact that is not a widespread technology means that it requires some training to adapt to the HTC Vive controllers (2014, p. XI).

Nonetheless, the metaphors of Ubiquitous Music Systems are translated into VIVR as the environment is the instrument, a space to be freely explored by the users; musical gestures can be rearranged in time by the concept of looping; the spatial tagging metaphor is intrinsic to VR, virtual elements are used to enhance interaction, hence creating a strong relationship between contextual elements and the creation (2014, p. XVIII). The Ubiquitous Music Systems' design patterns used in VIVR are: usability for both musically untrained participants and musicians; use of rapid prototyping and iterative design; application of the four musical interaction patterns: natural interaction, event sequencing, process control and mixing (2014, pp. 27-35)

An important part of examining the design of VIVR was a series of user tests. The first prototype of VIVR was demoed at Aalto's Department of Media's *Demo Day* in spring 2018. The second prototype was tested during the September of 2018 at Aalto Studios. The first demonstration of VIVR was at a public event where the audience came to test the instrument. After a general description of the system and how it works, users were given free rein to test the instrument. An informal interview with each participant took place after each test. The second prototype was implemented after discussing users' feedback from the demonstration of the first prototype. This last prototype was tested in pre-arranged user tests in a controlled environment. Fourteen participants each took a half hour test that was divided into five parts. In the first part, the participant was introduced to the controls of VIVR. The second part aimed to familiarize the participant with the instrument under short free improvisation. Next, the participant was given the task of creating a short composition following a given form. This task was followed by a spoken interview with the participant to collate information on their experience of using VIVR. Finally, the participant filled out a questionnaire with the purpose of rating the usability and features of the instrument. While the findings of the second round of user tests were not implemented at the time of writing this thesis, they remain as a discussion and future ideas to keep in mind in the further development of VIVR. The following sections expand on the methodology introduced in this chapter.

4 VIVR: the design process

The VRMI implemented during the time of this thesis is part of VIVR (Vibrating Instruments in Virtual Reality), an ongoing research project at the SOPI research group at Aalto University. The aim for this thesis is to provide a working implementation that can demonstrate how VR can be a valuable platform for New Interfaces of Musical Expression (NIME). This section will examine how some of the features of VR, such as immersion, 3D audio, spatial interaction, bodily interaction, audiovisual feedback and shifted paradigms can benefit individual music interaction. This analysis will be done by providing a chronological overview of the design process.

On the technical side, the hardware of the instrument consists of an HTC Vive, the virtual environment is created through Unity 3D and SuperCollider is the sound engine. Communication between the HTC Vive and Unity 3D is done through the SteamVR SDK. While Unity 3D and SuperCollider communicate via OSC. These technologies were selected because their suitability for the project. The HTC Vive provides a broad range of user interaction thanks to its camera tracking system. Unity 3D provides an easy development environment with fast prototyping. Last but not least, SuperCollider's division between server and language provides a flexible and powerful audio stream to work as the exclusive sound engine where real-time synthesis, ambisonic encoding and decoding, as well as OSC communication can happen all at the same time. A first-person video demonstration is presented to support this work[3].

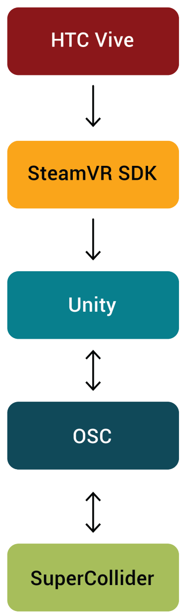


Figure 1: Communication pipeline

[3]https://vimeo.com/272217068

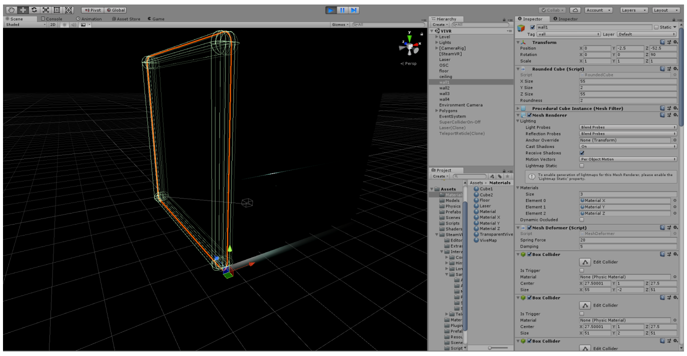


Figure 2: A procedural 3D wall

4.1 Creating the environment

When conceptualizing VIVR, one of the first VR features that SOPI wanted to target was immersion. According to Slater (2009, p. 3550), immersion can be understood as the sensorimotor contingencies that a system supports. These contingencies are a set of logical behaviours that are relevant in terms of perception within the virtual environment represented. To partially fulfill this objective, VIVR places the user inside the instrument rather than creating a 3D modelled instrument, externally placing the user. Thus, the instrument is actuated internally. This decision not only brings a superficial level of immersion as the user is surrounded by the instrument. As it will be demonstrated in this section, it also makes the user a factor of the instrument.

When the user enters VIVR, the environment feels like a room. A set of six surfaces form a cuboid with the user originally placed in the center of the floor. The decision of a cuboidal shape was made because it provides the most simple set-up. This simplicity is relevant as it provides the user with a blank canvas that can be totally deformed as music interaction develops. The walls of the cuboid are custom 3D models based on the combination of two tutorials made by Jasper Flick[4]. The Rounded Cube tutorial explains how to make rounded cubes based on vertices arrays that can change shape according to the parameters of X, Y, Z sizes and roundness. One can get a procedural 3D shaped wall of 55 units by setting these parameters in the Unity Editor to X Size = 55; Y Size = 2; Z Size 55; and Roundness = 2. Each wall game object holds an instance of the RoundedCube.cs and the MeshDeformer.cs scripts.

[4]https://catlikecoding.com/unity/tutorials/rounded-cube/
https://catlikecoding.com/unity/tutorials/mesh-deformation/

Using the RoundedCube.cs[5] script inside the Mesh Deformation tutorial instead of the Cube Sphere proposed by the author, it is possible to deform the walls according to a force input. When the force input hits the vertices that form the custom 3D shapes, it displaces them giving place to deformed shapes. The core scripts of Mesh Deformation are the MeshDeformer.cs script attached to the game object that one wants to deform, and the MeshDeformerInput.cs attached to the game object that is intended to be the deformation source. The MeshDeformerInput.cs script has two parameters to control: the force applied to the deformed object and a force offset to make the deformation follow the direction of the force input. On the deformed object, the MeshDeformer.cs script has public parameters of Spring Force and Damping. Spring Force sets how big the leap of the vertices is while jumping back and forth. The Damping parameter sets how smoothly this bouncing happens.

As for the user's game object representation, a CameraRig parent game object provided by the SteamVR SDK package is used. This parent holds three children: Controller (left), Controller (right) and Camera (head). From the Camera (head), only the Camera (eye) child is used as SuperCollider is the exclusive sound engine. Each of the controllers holds its own MeshDeformerInput.cs script with the force variable coming from the FFT values of the sound sources. The setup for this communication will be expanded in the following subsections. Additionally, each controller is represented by a 3D model of the HTC Vive hand controllers.

In order to decrease motion sickness, the user is able to teleport from one part of the environment to another by pointing at it and pressing the trigger. Teleportation results in a sudden change of spatial location while the user's body remains static. Compared to vection, where the visual environment moves gradually and the user is stationary, teleportation's sudden change do not confuse the user's perception as far as user's yaw orientation remains the same (LaValle, 2017, p. 232, 291). The teleportation function can be found under the listing (2).

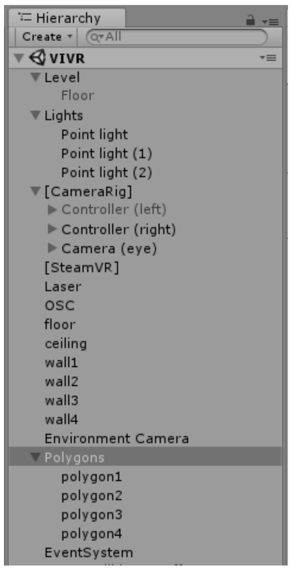


Figure 3: Game objects hierarchy in Unity 3D

[5] The modified version of this script can be accessed in listing (1)

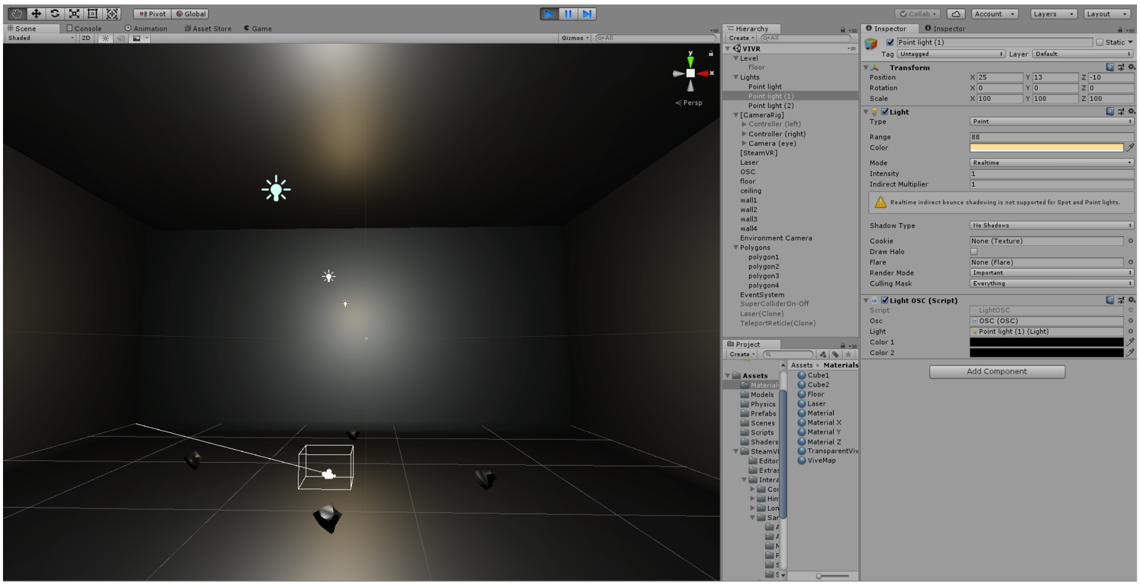


Figure 4: The virtual environment

With respect to the dark atmosphere
of the cuboid, this was an artistic decision. A dark environment helps represent the unknown. In this case, the inside of the instrument, an unusual place for the user that brings a new musical experience. This setup was made by applying black materials to wall game objects and a set of three point lights. These point lights are aligned horizontally with a white light at the centre a yellow light at one extreme and a light blue at the other.

4.2 Targeting feedback and mapping

The first design principle that Serafin, Erkut, Kojs, et al. (2016, p.26) mention is the design for feedback and mapping. This is understood as the relation and reaction of sonic, visual, haptic and perceptual elements together. Under this principle, different parameters such as envelope duration, envelope trigger, fundamental frequency, modulating frequency or buffer number of particular synthesis techniques like FM, wavetable and granular synthesis were mapped to the user's movement interaction in the environment. In the end, granular synthesis intrinsic features such as wide harmonic range, grain envelope, buffer reading position and envelope trigger rate showed to have more complementary mapping possibilities for VIVR's case than the other two. The following paragraphs will expand on how and with what purpose these parameters were mapped.

For the sound synthesis implementation of VIVR, a modular design of the audio engine seemed to fit better the iterative methodology of this project. Consequently, new sound modules are added as new features are required. This also allows keeping some connections between modules to remain intact, while changing others. Moreover, this decision allows to design one module and create as many instances of it as desired. Thus, each controller holds an instance of the granular module. Listing (3) shows the modular set-up and connections of the system, while listing (4) displays the module used as the main sound source for each controller.

In order to control these module's instances, a communication protocol between Unity 3D and SuperCollider is needed. Open Sound Control (OSC) provides a fast and powerful method of exchanging message bundles from a server to a client over a network. In this case, the bundles are being sent over the localhost address because Unity 3D and SuperCollider are running on the same workstation. OSC is built within SuperCollider as it is how the audio server and language communicate with each other. As for Unity 3D, a script written by Thomas Fredericks[6] allows this communication.

Concerning the control of these modules, natural interactions were taken into consideration. For example, if a movement is slow and steady, the sound should have a long envelope; if the movement is fast and has a sharp end, the sound should be percussive; if the movement is fast but constant, then the sound should grow in sonic layers. Therefore, the interaction design for these two granular modules is strongly based on the user's movement acceleration. This calculation allows measuring how fast the controllers travel from the original coordinates to the next ones, thus making it possible to categorize different kinds of movements. A cohesive mapping of these movements to sound is then achievable by scaling the acceleration calculation to different values according to the parameter targeted. Listing (5) shows how velocity and acceleration can be calculated in Unity 3D and then sent via OSC. As Unity 3D gives the velocity and acceleration of the 3D Cartesian coordinates, the average of these is calculated in SuperCollider as shown in listing (6). The one-to-many mapping of this new value is displayed in (17).

Once both instances of the Granular module are controlled via Unity, it is possible to gather data from a spectral analysis of the sonic output. In order to do so, the output of the sound source is routed to two different Fast Fourier Transform (FFT) modules and an amplitude tracker. The first FFT module[7] reads the frequency data

[6]https://github.com/thomasfredericks/UnityOSC
[7]This module is based on the one implemented by Fredrik Olofsson `https://sccode.org/1-4Wt`

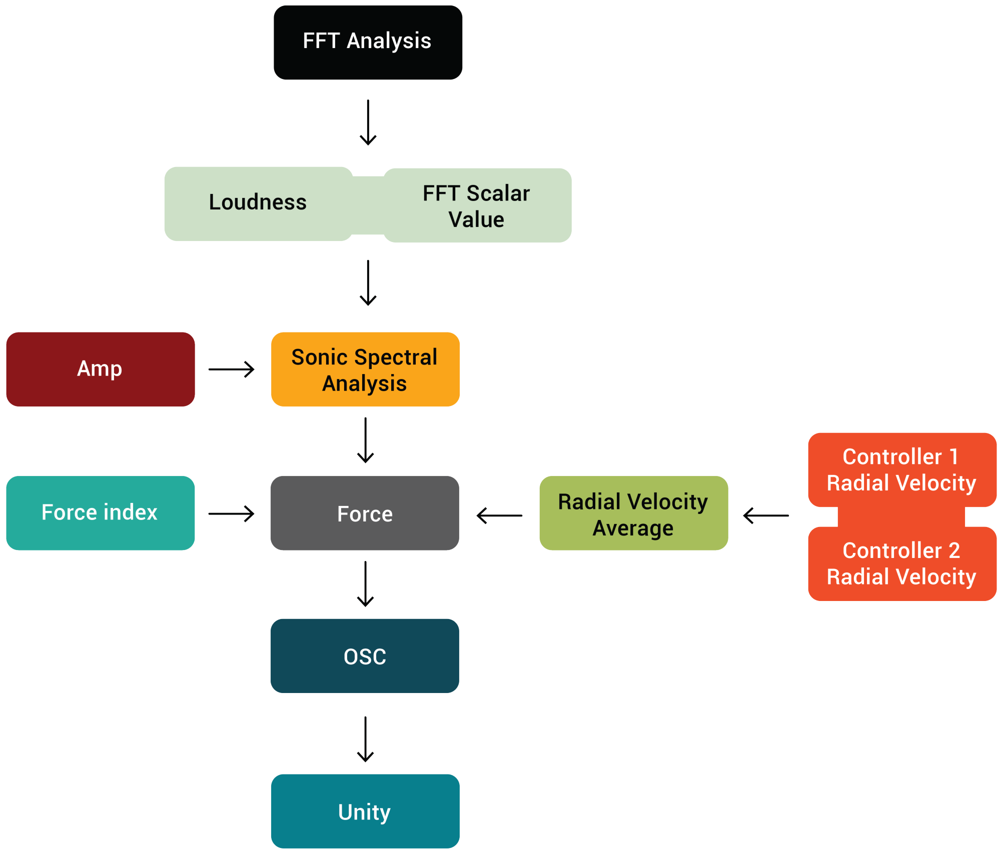


Figure 5: Force calculation

in relation to the time domain, meanwhile the second reads the perceptual loudness. The amplitude tracker, as its name implies, analyses the fluctuation of the sound wave. The values obtained are multiplied to get a total magnitude of the spectral analysis. However, in order to keep a consistency between user movements and sonic output, the squared magnitude of the controllers in relation to the user are calculated.

This calculation gives information on how far the controllers travel from the user. From the results, one can gather the direction of the movement and, if multiplied by the acceleration, the intensity of the movement. The following listings (7), (8), (9) show the spectral analysis modules and the listing (10) displays how the total magnitude of the analysis for each controller is sent to Unity 3D. Last, in Unity 3D, these values are applied as the force to the hit object, thus deforming it. Listing (11) presents how the values of the spectral analysis are obtained in Unity 3D and applied as the force to the hit object.

With regard to design for feedback and mapping principle of Serafin, Erkut, Kojs, et al. (2016), this relation between sonic output, user movement interaction and visual reaction turns out to be valuable. During the implementation of this relation, it was evident that a meaningful connection can only be achieved by working on these three elements as a whole. Regarding the haptic relation, it was left aside on the grounds that HTC Vive actuators are based on a rumble system, hence one can only set a particular rumbling time and trigger it on every sound event. As a result, this implementation was more distracting than immersive. In the following section, 3D audio's role in targeting the perception relation will be explained.

4.3 3D audio: extending the spatialization element

So far, this thesis has discussed that a 3D environment creates a concept of space and holds 3D objects within it, but this is not enough to engage in full immersion. In order to do so, certain real-life physics of motion and visual rendering need to be followed in order to trick the viewer's perception (LaValle, 2017, p. 51). Furthermore, in the case of a VRMI, there should be a feedback relationship between the control of the instrument and its audiovisual reactions. On the sonic level, 3D audio can expand on this sensory stimulation by creating a sense of space and by giving auditory cues of 3D structures and audio sources' location. This illusion is created by affecting the listener experience with the help of physical or virtual loudspeaker arrays that simulate an acoustic environment (Godfroy-Cooper et al., 2017). This section will explain the 3D audio implementation of VIVR and how it also acts as another musical element that brings a new layer of performance.

One of the most common techniques of 3D audio in VR is “Ambisonics". This is a 3D recording and playback method that based on sound field representation. Due to its features of encoding and decoding, Ambisonics offers flexible possibilities of sonic manipulations and spatialization in 3D environments with various types of set-ups (Frank et al., n.d., p. 1). In this technique, the original audio is encoded to a particular number of signals corresponding to the degree of accuracy that is wanted. These degrees of accuracy are called orders. These orders are representations of sound field excitation in terms of orthogonal basis functions or spherical harmonics. From the first order (4 channels), the level of accuracy, and thus the number of channels, is implemented by the formula $(N+1)^2$, where N is the ambisonic order (n.d., p. 2). In the case of an ambisonic three-dimensional representation, the encoded signal can be transformed in the distortion angle, azimuth and elevation parameters in order to change its perceptual location. As a final step (playback), the audio signal is decoded according to the array of loudspeakers that is used and its arrangement (n.d., p. 2).

The transformations of phase and gain that sound field representation applies to the encoded signal affect human's psycho-acoustic perception. Therefore, even though the signal is being played back in all loudspeakers at the same time, the sensation is that the sound source is located in a specific point in the space or moving across the sound field (Neukom & Schacher, 2008, p. 2). This fact makes Ambisonics differ from other methods such as vector-base amplitude panning (VBAP). In the latter case, amplitude levels rise on the nearest speakers to the virtual sound source and diminish on the furthest ones. Accordingly, VBAP methods lack the smoothness and surround quality of Ambisonics (Frank et al., n.d., p. 3).

Referring to the work of (Berkowitz et al., 2016), it is understood that 3D audio can become a music feature in two ways. One could give total control to the user in order to spatialize and rearrange sound trajectories, or the system can have some autonomy by spatializing sounds itself. In the case of VIVR, even both solutions were implemented, it was decided to emphasize the control that the user can have over sound spatialization. This decision came from the focus on user movement interaction. For example, if the user points to a location in the environment, as the environment deformation and sound processing are a result of that action, the expected reaction is that the sound source plays from that particular point. At the same time, when the user moves around the space, sound sources' playback has to be readjusted in order to represent the new spatial difference between the user and the sound source. In addition, user movement control allows creating sound spatialization trajectories in an intuitive way.

In order to set up the correlation between sound location and user position, it is necessary to extract the Cartesian coordinates of both. Inside Unity, one can get the user position in the 3D environment and head rotation from the Camera (eye) game object. Head rotation is important because it will give information on where the user is looking. Consequently, it is possible to translate the sound source according to the user's new point of view. Listing (12) shows how these values are extracted from Unity 3D and send via OSC. Regarding the sound source location, it is necessary to extract it from a Raycast hit point, as this one comes from the controller's pointing coordinates. In this case, the Raycast function gives information of the Cartesian coordinates and distance between the controllers and the point in space of the collider they are directed to. This function is implemented in the same MeshDeformerControllerInputL.cs script as the force-deformation. Listing (13) displays how to get these values from Unity 3D.

Inside SuperCollider, VIVR uses the Ambisonic Toolkit developed by Joseph Anderson [8]. The decision to use this SuperCollider extension and not others was because it has continuous support from the developers and the community and can run on the Windows distribution of SuperCollider. At the same time, this toolkit was developed with an artistic focus in mind. Therefore, it features different types of spatial transformations such as rotation, mirroring, direction, focus and push, and allows setting different microphone beamforming or direction patterns. The features of Anderson's toolkit have brought the interest of the research and artistic community and have been implemented in other toolkits by Grond and Lecomte (2017) and Mott (2017) that will see more development in a near future. It is worth mentioning, that using Ambisonics plug-ins inside Unity such as Resonance or SteamAudio was an option left aside. This decision would have required routing SuperCollider's audio signal via a virtual cable to Unity, which would have cause latency issues. At the same time, it would have created a rupture in the signal flow that would not have allowed for using SuperCollider's effect processing in the Ambisonics' chain.

In regard to the DSP module of the ATK for SuperCollider, it expects an input, an encoder, a transform function and a decoder. As the sound system is a pair of headphones, the decoder is a binaural one based on measured HRTFs from the IRCAM's database. Although the transformation functions provide a good ground to explore, as displayed in figure 6 another layer of sound processing is added by supplying a delay unit between the initial encoding and the final decoding. Following the chain: input (N[9] channel signal) -> encoding (W, X, Y, Z channel signal) -> decoding (4 channel signal) -> process (4 channel signal) -> encoding (W, X, Y, Z channel signal) -> spatial transformation (W, X, Y, Z channel signal) -> decoding (N channel signal), one can distribute the signal process spatially. This chain achieves an output with a vibrating deepness and sonic texture that is not possible to achieve by processing the signal before the encoding or after the decoding. Listing (14) shows a first implementation of the Ambisonic module.

Considering that Ambisonics is based on spherical harmonics, it is necessary to obtain Spherical coordinates of radial distance, polar angle and azimuthal angle out of the values extracted by Unity 3D. As previously mentioned, the Spherical coordinates of the sound source will need to be in relation to the user's position and head rotation. Listing (15) shows how to map the Cartesian coordinates and Euler

[8] `http://www.ambisonictoolkit.net/documentation/supercollider/`

[9] Where "N" is an arbitrary amount of channels.

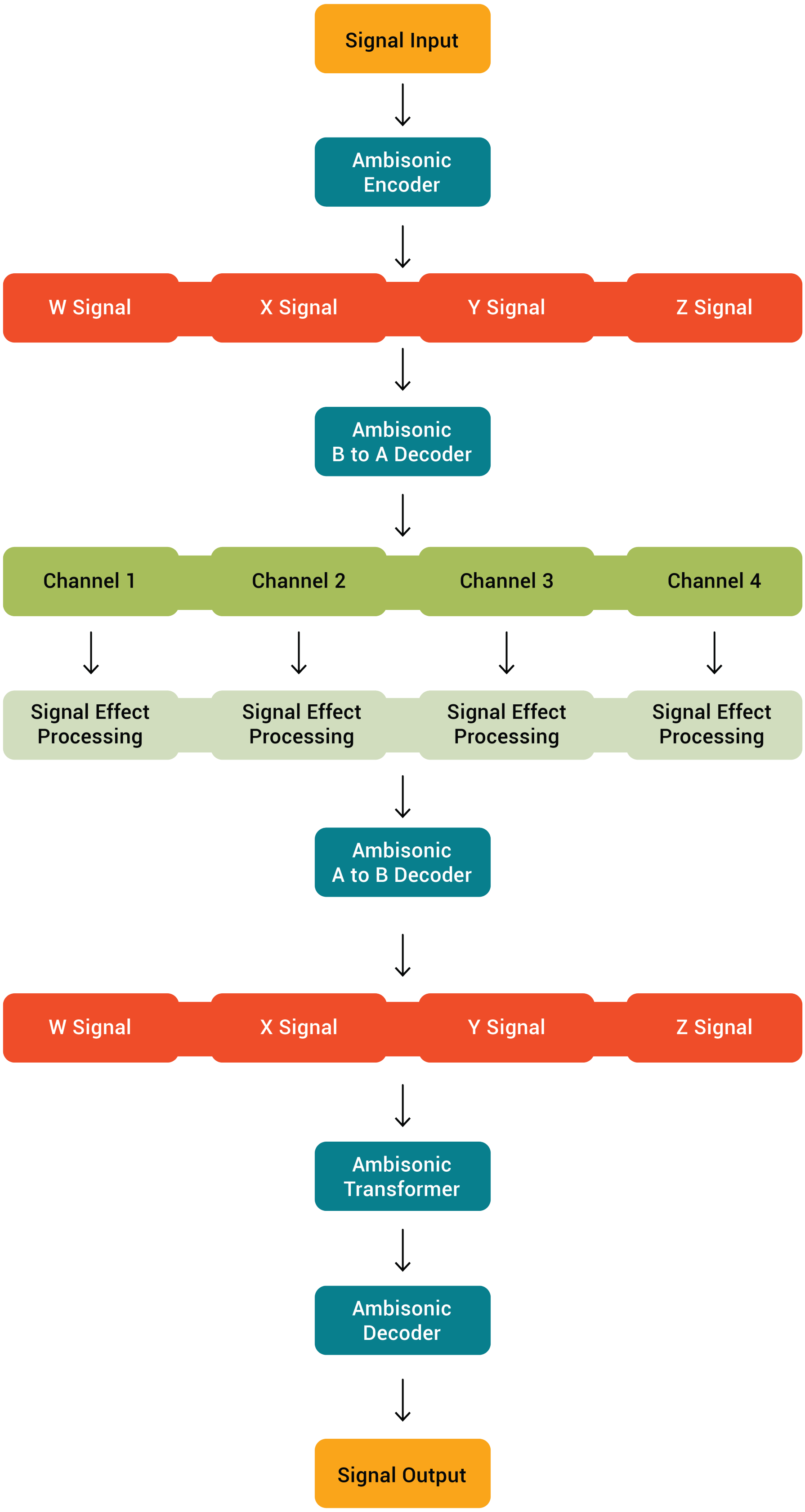


Figure 6: Ambisonics chain

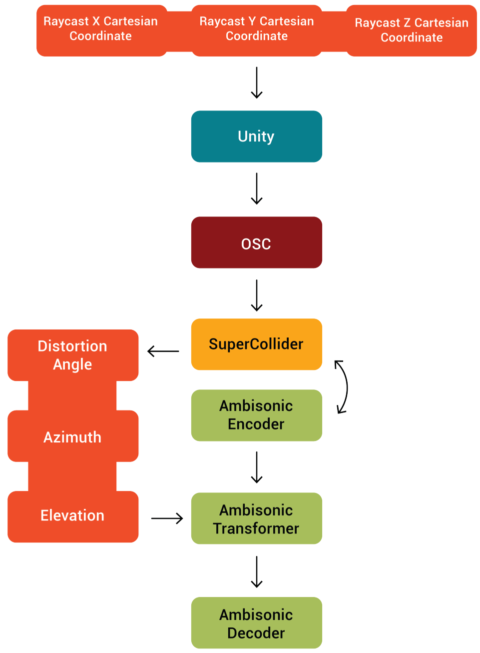


Figure 7: Cartesian coordinates to Ambisonic transformations

angles from the user to the values used in the ATK for SuperCollider. Listing (16) displays the formulas to obtain the Spherical coordinates and map them to the ATK for SuperCollider's values.

Consequently, it is necessary to map the Spherical coordinates to the parameters of the ambisonic transformation function. Thus, radial is mapped to distortion angle, azimuth to azimuth and polar to elevation. Moreover, some of these parameters can be mapped to the Granular module in order to expand on spatial interaction. In VIVR case, the buffer position of the granular reader depends on the distortion angle, and the grain size grows the closer the user is to the pointed collider. Listing (17) shows the mapping of these parameters. As the next step, it seems reasonable to route the output of the Ambisonic module to the Spectral analysis trackers instead of the Granular module as the final outcome will come from the Ambisonic module.

The implementation of the Ambisonic module as an artistic tool demonstrates how the creation of new sound fields through spatial transformations helps generate a better awareness of the narrative implication of sound spatialization in music. It is in the terrain of creation and transformation where one can find 3D audio's greatest artistic potential. How the 3D environment resonates and its sound components are processed in new ways are key features to enhance spatial interaction.

4.4 Towards a musical instrument: Expanding the sonic capabilities of VIVR

The implementation of the Ambisonic module adds the last factor to the basis of the instrument. In the previous sections, it has been shown how the user's movement interaction can generate sound, manipulate the sonic material and 3D structures of the environment and move sound around the space. As discussed by Tahiroğlu et al. (2018, p. 126), a musical instrument can be distinguished from a sound producing tool because a set of idiomatic gestures have been developed for it. Founding on this statement, a wider set of sonic capabilities will allow differentiating more easily among different sonic structure or sound events. Consequently, a cohesive mapping of particular sonic elements to specific actions helps in developing a collection of playing techniques that will help develop a performance practice (2018, p. 129). Therefore, to bring VIVR to a new level of instrument design, it is convenient to create new sound modules that will extend its playing possibilities. The modules described in this section do not expand on the idiomatic vocabulary per se. Rather, they constitute a new set of features that have to be explored by the user in order to create musical events that will develop the performance practice.

4.4.1 New sound processing layers and control possibilities

The first approach that was designed to control the instrument's amplitude was completely based on the user's movement velocity. This approach was thought resembling the physics of an acoustic instrument. A slow movement would produce a low in intensity sound, meanwhile, a fast one will produce a loud sound. After demonstrating a prototype version of VIVR, it was obvious that this feature makes performing the instrument an intensive physical task. This was due to other parameters also being based on controllers' acceleration. User's feedback asked for an easier control of the amplitude. The solution to this problem was found out by mapping on/off of the amplitude to the touch of the controllers' trackpad. The decision of this button was made upon the ergonomics of the controllers where the thumb finger usually rests on the trackpad. When touching the trackpad, amplitude levels rise to a minimum and from this point, the intensity grows according to the user's movement's acceleration.

One of the ideas that were brought up during a design discussion was the possibility to freeze the environment. In this sense, the 3D structures that are constantly deformed by the user will be suspended in their current state giving place to a new spatial environment. Sonic wise, the fairest representation of this feature was an FFT Freezing module. In both Unity 3D and SuperCollider, freezing was mapped to the grip buttons of the controllers. This mapping seems coherent considering that one has to apply more force when holding the controllers thereupon stopping the current deformation. As shown in listing (18), freezing the 3D structures' deformations can be done by calling the spirngForce component of the MeshDeformer script and setting it to 0 instead of 20. Listing (19) shows the DSP implementation of this freezing module and listing (20) how the parameter is controlled via OSC. As seen in listing (20), because the output levels varied when the Freezing module was activated, it was necessary to implement a function that would boost the output levels slightly when this module was activated.

In addition to freezing the state of the environment, another quality that it is intrinsic to sound and movement is pitch shifting. Having the Doppler[10] effect as a reference, it makes sense that there is some connection between the sonic manipulations moving around the space and their pitch perception. Nevertheless, in order to give the user another creativity tool and not to base VIVR entirely in real-life physics, a pitch shifting module with distortion was mapped to the trackpad Y axis. In this manner, high pitches were set to the upper middle of the axis, meanwhile low pitches to the lower middle. The distortion gain depends, once more, on the user's movement acceleration, the higher this one is the more gain it applies to the distortion chain. This module was set in between the Freezing module and the Ambisonic module, giving the possibility of manipulating the sonic input even when it is frozen. Listing (21) shows the DSP implementation of this module. Listing (22) displays how the trackpad touch and its Y axis are collected in Unity 3D to be sent to SuperCollider. Listing (23) represents the SuperCollider mapping of the trackpad values.

4.4.2 Autonomous actions and entities' relation

So far, it has been explained how the correlation between audiovisual feedback and bodily interaction positions the role of the user with the following factors: new control possibilities, audiovisual experiences, 3D sound spatialization and spatial interaction. These new challenges can be used to provide more exploratory approaches to music making. As discussed by Berkowitz et al. (2016), designers can set different levels of instrument's autonomy in music interaction. On the same topic, Tahiroğlu et

[10]the Doppler effect is the perceived change in the wavelength caused by the movement relation of the sound source and the listener between each other (Bažec & Dimc, 2018)

al. (2016, p. 446) defend that instrument's own music actions engage better the performer when they are designed systematically and are not completely random.

In accordance with the previous ideas, different environment autonomous responses that aim to support music interaction were designed for VIVR. As previously mentioned, the acceleration of each controller's movement is calculated and mapped to different sonic parameters. With the aim of detecting whether the user is engaged in the performance or not, an average of both controllers' acceleration is also calculated. A constant rate of high values on this average calculation determine a good level of engagement, meanwhile, a rate of low values during a certain amount of time resolve in low-level of engagement. According to the results of this calculation, the system responds in three different layers.

When the user is not active and this lack of activity produces a silent moment that is long enough to not be considered a part of the musical form, this condition instigates the environment to start playing on its own and reacting visually to its own sounds. In contrast with the sharp sonic gestures that produce energetic deformations, the instrument induces the user to restart the music activity with an evolving wide range sonic texture. In order to implement this autonomous reaction, when the average acceleration values are under the threshold of active movement, a counter of ten seconds is triggered. When it reaches this time, a 3D capsule game object is activated. Listing (24) displays how this measurement is implemented. The mesh renderer of this game object is deactivated, so the object is not visible. The 3D capsule travels around all the possible coordinates of the 3D space under the control of a synthesizer module that holds one sine LFO and two random LFOs. The amplitude values of these LFOs are mapped to the spatial limits of the environment and sent to Unity via OSC, as seen in listing (25). In the same way as the controllers, the 3D capsule holds a MeshDeformerItself.cs script. A calculation of the FFT output of the Ambisonic module that represents the environment is sent as the force to the MeshDeformerItself.cs script from SuperCollider. In order to deform the environment according to where the 3D capsule is pointing to, a Raycast hit is calculated from its transform position and its pointing forward direction. Considering that the speed of this 3D capsule is constant, the output values of the Raycast hit calculation are mapped to the parameters of both the Ambisonic module and the Granular module, as displayed in listing (26). The invisible capsule's implementation of deforming and sound generating features of the invisible capsule misleads the user's perception into thinking that the 3D structures of the environment deform themselves and consequently create a sound output.

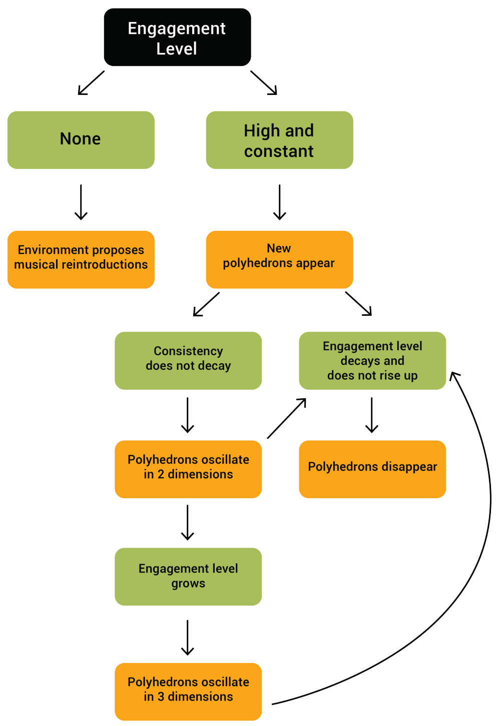


Figure 8: Instrument's autonomous reactions

On the contrary, when the user engages in a persistent activity above the threshold, the environment responds with the creation of a new level of procedural meshes. These new 3D Structures hold four different instances of the Granular, Freezing and Ambisonic module. The activation of this new layer follows the logic of the environment's response mentioned above, but vice-versa, as shown in (27). These new 3D structures are a set of four polyhedrons that are placed symmetrically at different points of the environment's center. As each of these polyhedrons holds a Granular module, the sample of each module is different. The new addition of these modules allows more diversions of the sonic footprint during the performance, hence making it more likely to build up different music sections. The last response from the environment comes when the polyhedrons are activated and the user's high-level of activity keeps constant. After a particular amount of time, the polyhedron's set starts oscillating horizontally. If after that same time section the level of activity is over a second higher threshold, the set oscillates in the three dimensions. Listing (28) displays how this set of oscillations are triggered. Considering Ambisonics' transformations and spatial mapping of sonic parameters, these oscillations become significant when developing new sonic interactions.

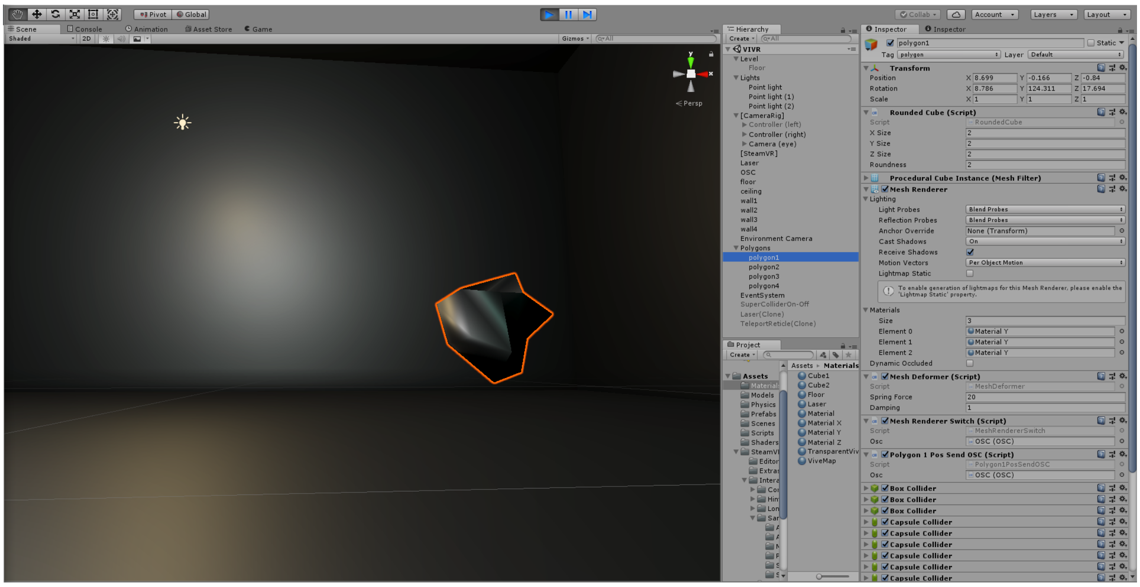


Figure 9: A polyhedron 3D mesh

As the walls that form the environment, each polyhedron is a game object that holds a RoundedCube.cs a MeshDeformer.cs and a PolygonPosSendOSC.cs script. In order to create the particular shape of these polyhedrons, the RoundedCube.cs parameters of X Size, Y Size, Z Size and Roundness are set to "2". The deforming force for these polyhedrons also comes from the FFT analysis of the Ambisonic output module. Nevertheless, after this new addition, the system needs to know which of all the new objects the controllers are pointing to. This categorization is possible by extracting the game object out of the RaycastHit.collider.gameObject method in Unity 3D, as represented in (29). The same information is sent to SuperCollider in order to control the different instances of the Granular, Freezing, and Ambisonic module. Once the collider type is gathered in SuperCollider, it is possible to control the parameters of the instances that belong to the pointed collider. Listing (30) shows how the categorization is set in SuperCollider. The polyhedron's modules' parameters are not only mapped to their position in space but also to their distance to the user, as seen in listing (31).

4.4.3 User's feedback based expansion of the instrument

The first prototype of VIVR was showcased to the public for the first time in spring 2018 at Aalto's Department of Media *Demo Day.* The audience of the event tested the project and gave feedback afterward. From all the input, people generally agreed that the possibility to loop music actions would improve the musical experience. Another common suggestion was the addition of a function that would change both controller's samples to truly contrasting ones. Listing (32) enumerates the samples used in each Granular module after this suggestion.

The idea of implementing a looper module felt coherent in the sense that it would allow the user to create a music layer that will support new music material. At the same time, loops can be used in different sets of frequencies, giving the possibility to play different music registers at once. Finally, one of the simplest music forms "ABA" is based on repetition. Looping allows bringing back previous music elements and creates a musical structure by iteration. In VIVR, this looping feature is implemented by adding a new Looper module per controller. Each looper is constantly recording new material over four seconds. As the trigger of the controller is pulled, the Looper module is activated and the Granular module deactivated. The loop runs as long as the trigger is pulled. After testing this new module in VIVR, one of the most interesting outcomes was the possibility of re-spatializing previous music segments that were played in different locations before. This feature is another enrichment that VR and 3D audio can bring to music interaction. Listing (33) shows the DSP module and its activation function. On the Unity 3D side, this module is activated as shown in listing (34).

Fulfilling the idea of changing to a contrasting sample could be easily done by changing the buffer index with a button action. Nonetheless, at SOPI it was discussed that applying spatial interaction to this feature could create a notion of zones within the environment. Moreover, this feature could be extended so that these two contrasting samples would be exponentially interpolated according to the user's position in the space. As a result, jumping from one zone to another would not just change the sample, but the areas around the zones' borders would be a mix of both contrasting samples. This factor is made by creating two more instances of the Granular module for each controller and routing their output to an xFader module per controller that interpolates the now two Granular modules of each controller. Listing (35) displays the DSP module of the xFader and how the crossfade parameter is mapped to the X transform.position axis of the user.

This chapter has described how different ways of interaction can be implemented to develop music interaction. In this manner, the particular controls and technological capabilities of VR are used with a musical purpose in mind, and not just for the sake of using them. In a broader perspective, specialization and categorical delineation are strong within computer music research. In this category, developments of musical practices are partitioned by their related musical technology and separation of the performer, instrument and environment (Waters, 2007, p. 2). On the contrary, considering entity relationships and factoring common features in VR environments, the lines between these factors get blurred. As it has been shown, in order to expand on the music capabilities of a VRMI, all these actors become active agents that feedback into each other through means of musical content and interaction. Thus, when developing a VRMI, one should think of the user, not as the only actuator of the instrument, but as an entity, a design factor that belongs to a performance ecosystem (2007, pp. 4-5). In this manner, it is the relation of interactions between the environment and the user together that drives the musical content, but not any of them separately.

5 User tests

The previous chapter described the design approach and factors taken into consideration when planning a coherent use of VR's affordances in a musical context. Even though section 4.4.3 of the previous chapter refers to the new features implemented after user feedback, these tests were based on a two to three-minute free improvisation with VIVR and were not documented. Due to the brief feedback obtained and the lack of documentation, new user tests took place in a controlled and documented environment in order to validate the degree of usability and capabilities of VIVR. This chapter describes the methodology used in the user tests and analyzes their outcome.

These user tests were made in fourteen individual sessions of half an hour each. The structure of the test was divided into five parts: an introduction to the control of the instrument; a three-minute free improvisation; a two-minute composition; a spoken interview; a rating questionnaire. The first part introduced the instrument controls to the user. The user wore the HTC Vive headset and held the controllers meanwhile was told the different features of the instrument, the influence of bodily and spatial interaction and the function of each button of the HTC Vive's controllers. The second part consisted of a three-minute free improvisation where the user could test the different features and control of the instrument. In the third part, the user was given the task of creating a two-minute composition following the form ABA'. This task did not evaluate the user's musical knowledge but examined the diversity of musical attributes of the instrument. The fourth part was a spoken interview with the user in order to understand the experience outcome of using VIVR [11]. Finally, the user test concluded with a quantitative questionnaire in which the user rated different features of the instrument from one to five [12].

Regarding the profile of the participants, it was diverse. Thirteen users had had some previous experience with VR before, but not everybody was an experienced musician. This decision was made to test if this approach to VRMI's design would provide any benefit to a non-experienced musician. The following subsections describe the findings obtained from the interview and an analysis of the results from the quantitative questionnaire.

[11] The questions that drove the interview are listed in the appendix B. The interview recordings can be requested by contacting the author of this thesis.

[12] A copy of this questionnaire can be accessed in appendix C.

5.1 Qualitative findings

These findings are obtained from spoken interviews with the participants of the user test. Each interview lasted from seven to ten minutes and it was conducted in a casual conversational manner. The purpose of this interview was to study the overall experience of using VIVR.

Although each participant had different answers, everyone agreed that VR can be used as a musical tool. The general thoughts after trying the instrument varied according to each participant. Adjectives used to describe the experience were abstract, contemporary, different and exploratory. Playing the instrument spatially from inside proved to be a challenge to all participants. Eight participants acknowledge that this approach is very exploratory and therefore, it requires more time to discern the diversity of sounds that the instrument can produce. In relation to the sonic output of the instrument, eleven participants felt that it was designed with a target group or music genre in mind, which in this case was the NIME community. This assumption seemed to be conditioned by the chosen samples of the Granular modules. The fourteen participants concede that the current selection of samples affected the music they created. Twelve participants suggested that implementing a functionality that allows choosing from a different set of samples selected by the user, would help organize their musical ideas. However, six users found it difficult to generate a wide range of dynamics. As a result, creating the musical form ABA' was possible, but required some effort to find different levels of sonic density. Three participants proposed the implementation of several loop buffers so the user could build up different musical layers, bringing more musical diversity to the composition.

As a result of this test, one can discern that VIVR offers a new approach to making music based on bodily and spatial interaction. Nevertheless, the novelty of this approach requires time to learn it as is not as straightforward as pressing one key and getting a musical note. In this aspect, the wide spectrum of each Granular module focuses the music more on the timbrical aspect than in the melodic aspect. This direction can be unintuitive for users that are not familiar with experimental music making. At the same time, it is evident that the samples selected by the user have a big influence on music making. Thus, in a new version of this instrument, it is necessary to implement a feature that in real-time, would allow the user to select among different sets of samples. This new implementation would have an influence on the user experience of the environment, as three users indicated that the current set of samples created a particular atmosphere. Additionally, the mapping of sound dynamics needs to be revised as it represented an issue for the majority of the users when developing a musical narrative. The idea of including more loop buffers is a common technique in contemporary music creation and would benefit musical form development. In a broader perspective, these interviews confirm VIVR provides a new musical experience. Still, this new approach needs more development in some of its features such as sample sets, looping and dynamics in order to favour individual music interaction at a completely new level.

5.2 Quantitative findings

The quantitative conclusions presented in this subsection were obtained from an online questionnaire that the participants filled as the last part of the user test [13]. This questionnaire took inspiration from the comparison of interaction profiles by Lenz et al. (2013, p. 129). In this publication Lenz et al. (2013) measure interaction attributes on a scale from one to seven by confronting adjectives that describe opposite levels of interaction profiles. Similarly, VIVR's quantitative questionnaire provides different questions with a scale of one to five that quantifies the usability of the instrument and how its features enhance musical interaction.

The first part of the questionnaire evaluated the relation of the participants with VR technologies. Thirteen users had experienced VR before and an average of 2.57 represented how familiar they are with and how regularly they use this technology. As shown in figure 10, out of the thirteen users, only four of them had tried a musical environment or instrument in VR. An average result of 3.35 showed that the control mechanics of the instrument were easy to adapt to. However, when measuring intuitiveness, the result was 2.92. As presented in the qualitative findings, the adaptation to the control mechanics required some time to learn. In relation to latency, only one user found that the latency time made the instrument unusable. This same user felt motion sickness while none of the others did. An aspect that has been defended during this research is the new interaction possibility that Ambisonics and VR bring to music. The results of the questionnaire give an average of 4.07 when comparing the relevance of 3D audio with other musical elements such as melody, harmony or rhythm. The dynamic autonomous reactions of the instrument help to develop musical content with an average of 3.35. As mentioned by some users in the spoken interview, an average of 4.0 reckon that the feedback between sonic output and visual reactions was uniform. On the same level, an average of 4.07 shows that the mapping between movement interaction and audiovisual reactions was coherent.

[13] Figures with the results of this questionnaire are displayed at the end of this chapter.

Regarding immersion, as shown in figure 11 50% of the users felt that they were part of the instrument. In this sense, an average of 4.28 felt that experience was immersive. From the participants that felt the experience immersive, 57.1% agreed that this immersion helped them to engage more with the music they were making, as presented in figure 12. Finally, 78.6% of the participants did not notice the absence of haptic feedback, as shown in figure 13.

The findings of the quantitative questionnaire show that VR provides an immersive approach to music making and that this immersion can encourage music making up to a certain degree. In general, the results were favourable regarding VR affordances such as 3D audio, immersion and visual reactions. In this sense, the relation between sonic output and visual reactions seemed to be coherent enough for most of the users to not notice the absence of haptic feedback. However, the intuitive feature of the control mechanics needs to be improved, as also mentioned in the spoken interviews. At the same time, the system's autonomous reactions can be developed considering more conditional factors in order to bring a better connection with the user's own music making. Latency and motion sickness do not seem to be an aspect that will require further work. When running on a localhost, OSC communication between Unity and SuperCollider provides a fast enough exchange of data so as not to negatively affect music interaction. Finally, the percentage of participants who had not experienced a musical environment or instrument in VR before is considerable. Consequently, more research should be done in this area in order to study the different situations that VR can provide to musical interaction.

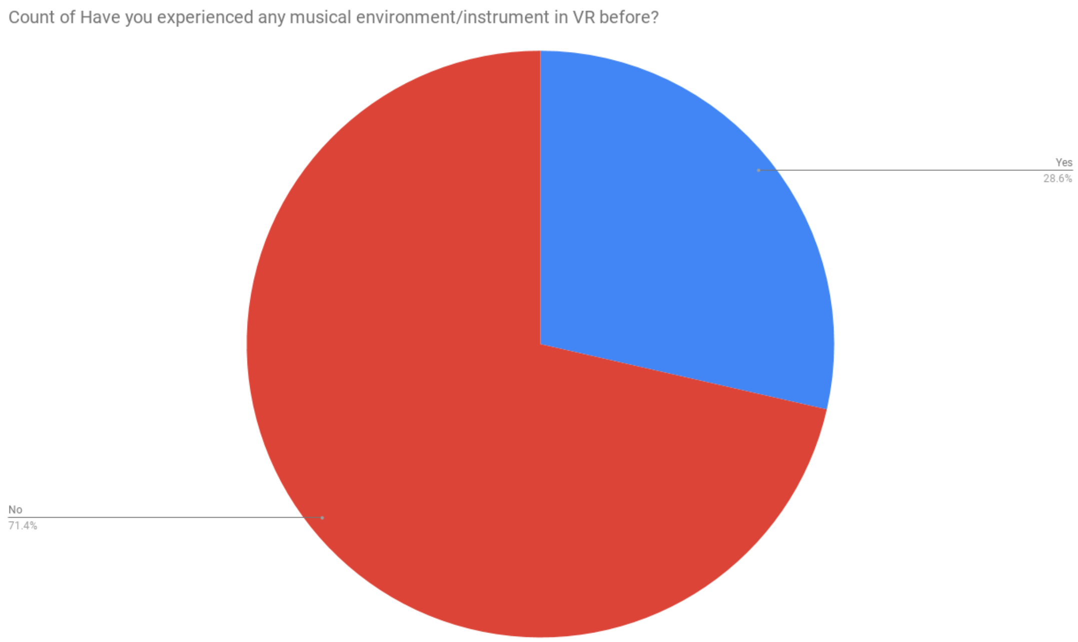


Figure 10: Percentage of users who had experienced a music driven experience in VR before

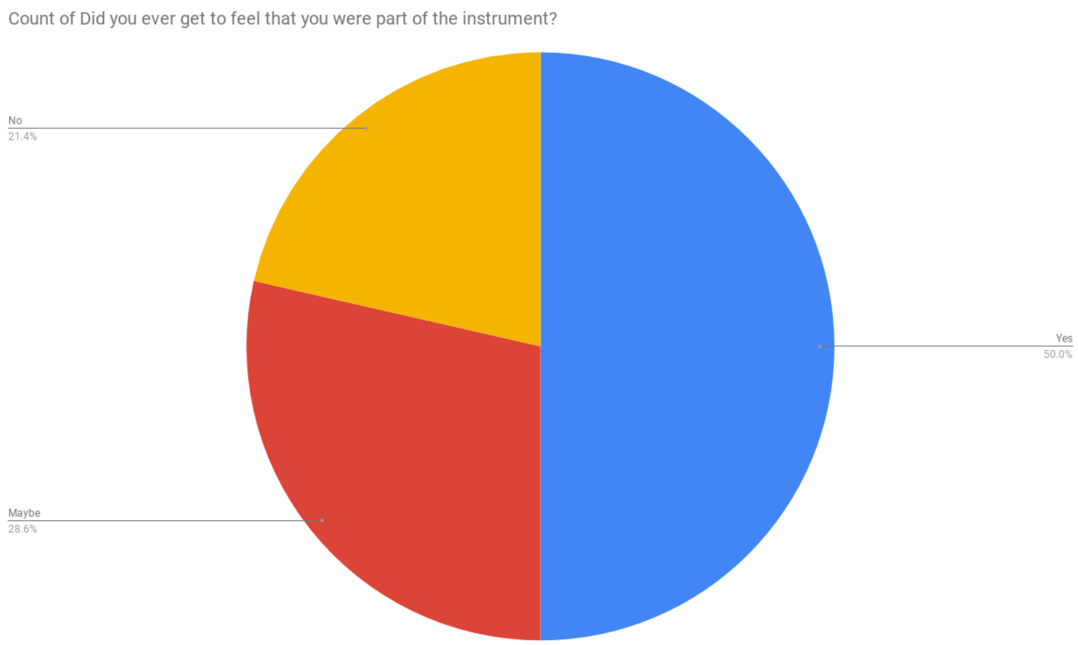


Figure 11: Percentage of users regarding their connection with the instrument

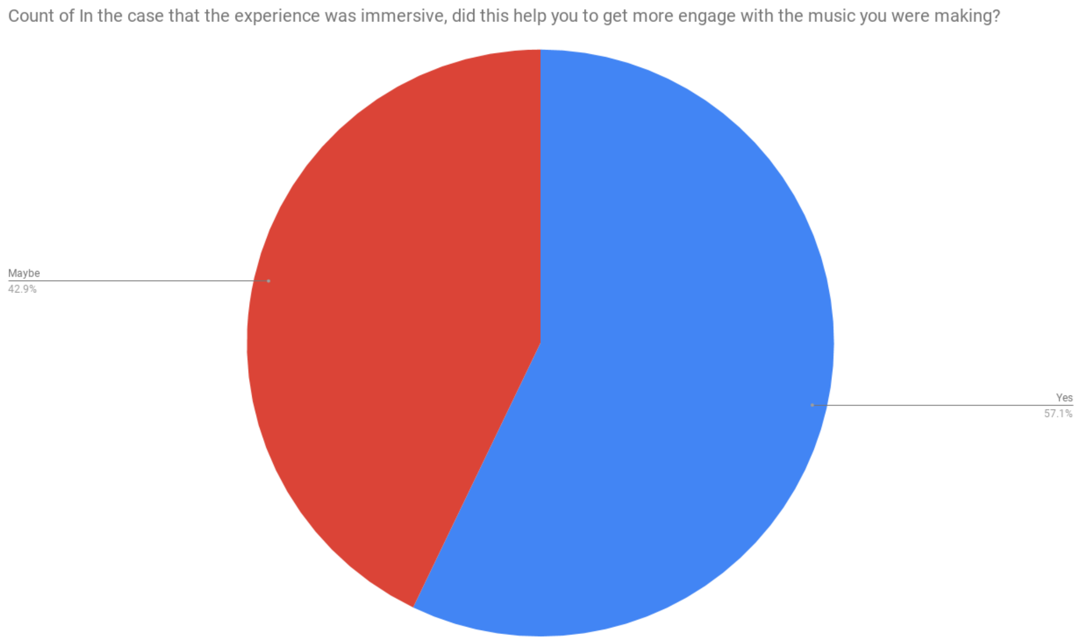


Figure 12: Percentage of users considering the benefits of immersion in music making

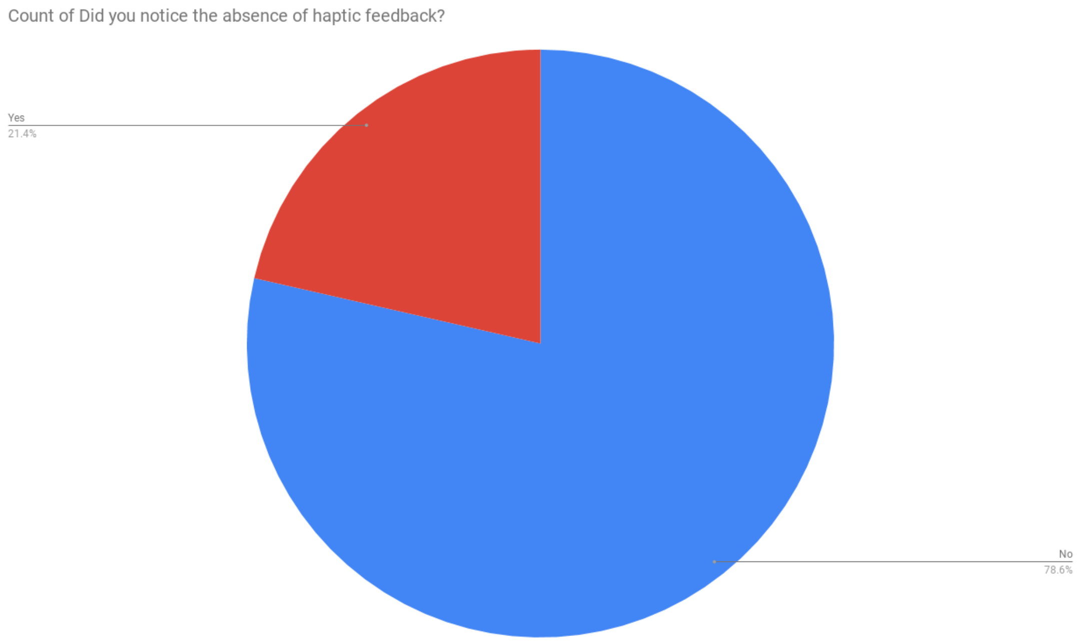


Figure 13: Percentage of users according to their observation of haptic feedback

6 Discussion

This chapter reflects on the contrast between user tests' findings and the design approach. The aim of this thesis is to research how an immersive VR environment can enhance individual interaction. This research question was approached by an aggregative design where the factors of 3D audio, audiovisual feedback, bodily and spatial interaction, autonomous responses and performer play equal parts in order to favour the musical content. The focus on an ecosystem of design factors has been the drive of this thesis. Echoing from the idea of "performance ecosystems" (Waters, 2007), music not only depends on the ability of the musician to play the instrument or the capabilities of the instrument itself. It is a complex reciprocal relationship network between the performer, instrument, audience, its social context and environment (2007, pp. 2-3).

As it was introduced in the section 1 of this thesis, the NIME community has been disbelieving the advantages that VR can provide to musical interaction. From the background work described in section 2, it is evident to see that VR's main feature to enhance musical interplay is immersion. However, immersion is not intrinsic to VR. It is a feature that is conceived by a combination of factors. The different VRMIs and design approaches in section 2 refer to audiovisual feedback, spatial mapping, sense of presence, an instrument's autonomous reactions and the individuality of the platform.

VIVR design approach to develop immersion and enhance musical interaction came from acknowledging what was done in those previous works and shifting the paradigm. The grounds for an ecosystem of design factors comes from placing the user in a different relationship with the instrument. The instrument is not anymore an external object within the spatial environment where the user is, but this environment becomes the instrument itself. Under this approach, the performance of the instrument becomes a spatial exploration of the surroundings.

Regarding the findings of the user tests, it is evident that this shifting of paradigm brings a new approach to music making. Nevertheless, a new playing technique requires time to learn and if it does not correspond to previous performance archetypes it becomes unintuitive. In this context, the one-to-many mapping relations of bodily or spatial interaction with different sound parameters become a challenge when targeting the same sonic output over and over again. Even though on the big picture the user is making the same gesture, unconsciously some variables such as acceleration, 3D coordinates or gesture range cannot be equal every time. In this sense, a combination of a less conditioned mapping with the machine learning approach of Deacon et al. (2017) seems to be the solution to follow.

As for the additional benefits that 3D audio can bring to music making, the user tests demonstrated that this is a research area to focus on. The results of the quantitative questionnaire put 3D audio interaction at the same level as other musical elements such as melody, harmony or dynamics. In this case, the mapping is more apparent as the 3D Cartesian coordinates are translated to Spherical coordinates, which are the foundation of ambisonic transformations. As the subsection 4.3 remarks, the ambisonic degree used in VIVR is FOA, thereby further experiments should take part when HOA gets implemented as part of the ATK for SuperCollider.

In relation to the audio functionalities of the instrument, the user tests indicate that the user should have a higher degree of control over the samples available for selection. The possibility of choosing from a wider range of samples in real-time would strengthen the music variation possibilities of VIVR. In the same category, the loop function should be implemented with more loop buffers. This would allow users to build up and access different musical sections, thus favouring musical creation.

The works of Lanier (1993) and Serafin, Erkut, Kojs, et al. (2016) refer to a multi-sensorial relation in tandem as a design factor to take into consideration. According to the user tests, there is a uniform relationship between the visual reactions of the environment and the sonic output. Even though haptic feedback was not implemented, only 21.4% of the users noticed the absence of it. Nonetheless, SOPI believes that this is a relationship that should be integrated into a future iteration of this project. A solution to consider would be to design a separate haptic system accordingly to other solutions that have been implemented by the NIME community.

With respect to the research question, an average result of 4.28 in the user tests illustrates that the experience of using VIVR was immersive. More precisely, 57.1% agreed that this immersion helped them to engage more with the music they were making, the other 42.9% expressed that perhaps it did, but none responded negatively. These results show the approach that the ecosystem of factors followed in this research helped enhance individual musical interaction. However, in order to combine a new platform with a new proposition to music making, a cohesive sense in every mapping dimension is necessary. In this context, a misleading relationship of factors can affect the overall interest in musical development.

Finally, VR is a medium that has not yet been established as a musical environment and it lacks a practice tradition. In order to establish an experience of VRMIs, development and performance should be constant, pushing their use towards an audience and iterating their design as this practice develops (Serafin, Erkut, Nordahl, et al., 2016, p. 270). In this latter aspect, the contribution of a music repertoire for VRMIs is necessary. Music history has shown that instrument design not only iterates because of technological improvements, it also evolves because new affordances and sonic characteristics are demanded by composers and musicians (J. C. Vasquez, Tahiroğlu, & Kildal, 2017, p. 175). In the end, the design of VRMIs is an ongoing process, and while it is convenient to settle some design principles, only use and performance will tell how practical these are.

7 Conclusion

This thesis aimed to answer how a VR immersive environment could enhance individual musical interaction. Following SOPI's VIVR project's methodology, this question was approached in different stages. The first step addressed the current situation of VRMIs in the NIME community. The second step studied previous ideas of VRMI design. Finally, this research was implemented in a working prototype. This approach was to create an ecosystem of factors where 3D audio, audiovisual feedback, bodily and spatial interaction, autonomous responses and performer were addressed in equal means and relationship. The research carried out in this thesis establishes that this approach is valid because it creates an immersive experience that allows for a better engagement with the musical content. However, this relationship of factors implies that the mapping of variables has to be equally coherent in each factor, in order to keep a constant commitment to the musical narrative.

The targeted group of this research is the NIME community. As indicated in section 1, there is a lack of research focused on VR within this group. The experimental and non-commercial development of musical content in VR platforms will not expand as far as new research groups do not show interest in the matter. As this thesis addresses, VR can serve as a means of enhancing musical interaction thanks to its affordances. Nonetheless, it requires performance and a repertoire in order to deepen the possibilities that it offers to music making.

The works of Lanier (1993) and Hamilton and Platz (2016) refer to the discrepancy between performing a VRMI and experiencing it as an audience. Thus, equally important design concerns appear when considering how this ecosystem of factors can be presented to an audience that does not possess the same influence in the alteration of the VR environment. This challenge denotes a limitation to focus on when exploring the balance between the performance ecosystem components. As for today standards, VR systems that support user tracking on room-based scaling are not affordable. As a result, further studies should be developed considering how to present VR musical experiences to a wider audience.

In conclusion, VR is a technology in a state of development that yet needs to be assimilated and acquired by the broader public. Nevertheless, the affordances of VR can contribute to creating new degrees of musical interaction. In order to collect new findings related to this matter, the design, repertoire and performance practice of VRMIs needs to be hastened. An established musical practice of VRMIs would bring more possibilities to showcase new forms of music making to an audience, thus experimenting what arrangements and contexts work best for the public.

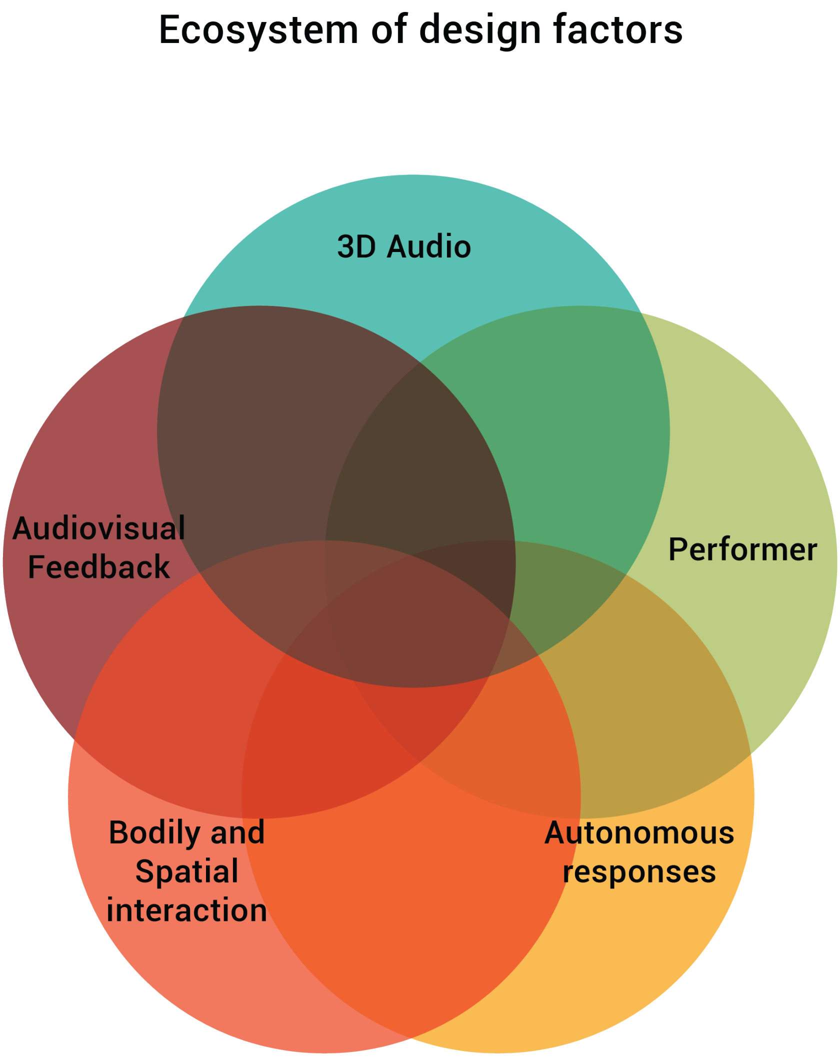


Figure 14: Ecosystem of factors

References

Bažec, M., & Dimc, F. (2018). Doppler effect measurement with sdr. In *26th international symposium on electronics in transport.* Ljubljana.

Berkowitz, Z., Berdahl, E., & Beck, S. D. (2016). Emphasizing Form in Virtual Reality-Based Music Performance. In H. Timmermans (Ed.), *Proceedings of the international computer music conference 2016* (pp. 341–344). Routledge. Retrieved from `https://quod.lib.umich.edu/cgi/p/pod/dod-idx/emphasizing-form-in-virtual-reality-based-music-performance.pdf?c=icmc;idno=bbp2372.2016.068;format=pdf`

Deacon, T., Stockman, T., & Barthet, M. (2017). User Experience in an Interactive Music Virtual Reality System: An Exploratory Study. In *International symposium on computer music multidisciplinary research* (pp. 192–216). Retrieved from `https://link.springer.com/content/pdf/10.1007/978-3-319-67738-5_12.pdf` doi: 10.1007/978-3-319-67738-5

Frank, M., Zotter, F., & Sontacchi, A. (n.d.). In *57th aes international conference.*

Godfroy-Cooper, M., Begault, D. R., & Wenzel, E. M. (2017). Perception of spatial sound. In *Immersive sound* (pp. 5–39). Focal Press.

Grond, F., & Lecomte, P. (2017). Higher Order Ambisonics for SuperCollider. In *Linux audio conference 2017 proceedings.* Retrieved from `https://musinf.univ-st-etienne.fr/lac2017/pdfs/13_C_D_141345N.pdf`

Hamilton, R., & Platz, C. (2016). Gesture-based Collaborative Virtual Reality Performance in Carillon. In *Proceedings of the international computer music conference 2016* (pp. 337–340). Utrecht. Retrieved from `https://www.researchgate.net/publication/318028813_Gesture-based_Collaborative_Virtual_Reality_Performance_in_Carillon`

Keller, D., Lazzarini, V., & Pimenta, M. S. (2014). *Computational Music Science Ubiquitous Music* (D. Keller, V. Lazzarini, & M. S. Pimenta, Eds.). New York: Springer. Retrieved from `http://www.springer.com/series/8349` doi: 10.1007/978-3-319-11152-0

Kubovy, M. (1988). *The psychology of perspective and renaissance art.* Cambridge University Press. Retrieved from `https://books.google.fi/books?id=aAc5AAAAIAAJ`

Lai, C.-H., Niinimäki, M., Tahiroglu, K., Kildal, J., & Ahmaniemi, T. (2011). Perceived physicality in audio-enhanced force input. In *Proceedings of the 13th international conference on multimodal interfaces* (pp. 287–294). New

York, NY, USA: ACM. Retrieved from http://doi.acm.org/10.1145/2070481.2070533 doi: 10.1145/2070481.2070533

Lanier, J. (n.d.). *Virtual instrumentation.* Retrieved 2018-07-15, from http://www.jaronlanier.com/instruments.html

Lanier, J. (1993). The sound of one hand. *Whole Earth Review*(79), 30–4.

LaValle, S. M. (2017). *VIRTUAL REALITY.* Cambridge University Press. Retrieved from http://vr.cs.uiuc.edu/

Lenz, E., Diefenbach, S., & Hassenzahl, M. (2013). Exploring relationships between interaction attributes and experience. In *Proceedings of the 6th international conference on designing pleasurable products and interfaces* (pp. 126–135). ACM.

Liang, R.-h., & Ming, O. (1994). Impromptu conductor:A virtual reality system for music generation based on supervised learning. *The Journal of DISPLAYS*, *15*(3), 141–148. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=72755241464947E6B6505C05F0B3122A?doi=10.1.1.586.508&rep=rep1&type=pdf

Mäki-Patola, T., Laitinen, J., Kanerva, A., & Takala, T. (2005). Experiments with virtual reality instruments. In *Proceedings of the 2005 conference on new interfaces for musical expression* (pp. 11–16). National University of Singapore.

Morreale, F., McPherson, A. P., & Wanderley, M. (2018, June). Nime identity from the performer's perspective. In T. M. Luke Dahl Douglas Bowman (Ed.), *Proceedings of the international conference on new interfaces for musical expression* (pp. 168–173). Blacksburg, Virginia, USA: Virginia Tech. Retrieved from http://www.nime.org/proceedings/2018/nime2018_paper0036.pdf

Mott, I. (2017). Botanica: navigable, immersive sound art. In S. Martins et al. (Eds.), *16º encontro internacional de arte e tecnologia: #16.art: Artis intelligentia: Imaginar o real* (pp. 526–533). i2ADS–Instituto de Investigação em Arte, Design e Sociedade. Retrieved from https://www.researchgate.net/publication/321579850_Botanica_navigable_immersive_sound_art

Neukom, M., & Schacher, J. C. (2008). Ambisonics Equivalent Panning. In *Icmc 2008 proceedings.* Michigan Publishing. Retrieved from http://decoy.iki.fi/dsound/ambisonic/motherlode/source/ICMC08_AEP_paper.pdf

Niinimäki, M., & Tahiroglu, K. (2012). Ahne: A novel interface for spatial interaction. In *Chi '12 extended abstracts on human factors in computing systems* (pp. 1031–1034). New York, NY, USA: ACM. Retrieved from http://

doi.acm.org/10.1145/2212776.2212378 doi: 10.1145/2212776.2212378

Serafin, S., Erkut, C., Kojs, J., Nilsson, N. C., & Nordahl, R. (2016). Virtual Reality Musical Instruments: State of the Art, Design Principles, and Future Directions. *Computer Music Journal*, *40*(3), 22–40. Retrieved from http://www.mitpressjournals.org/doi/pdf/10.1162/COMJ{_}a{_}00372 doi: 10.1162/COMJ

Serafin, S., Erkut, C., Nordahl, R., Nilsson, N. C., & Kojs, J. (2016). Virtual reality musical instruments: Guidelines for multisensory interaction design. In *Am '16* (pp. 266–271). Norrköping. Retrieved from http://dx.doi.org/10.1145/2986416.2986431 doi: 10.1145/2986416.2986431

Slater, M. (2009). Place illusion and plausibility can lead to realistic behaviour in immersive virtual environments. *Philosophical Transactions of the Royal Society B: Biological Sciences*, *364*(1535), 3549–3557. Retrieved from https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2781884/pdf/rstb20090138.pdf doi: 10.1098/rstb.2009.0138

Sutherland, I. (1965). The ultimate display. In W. A. Kalenich (Ed.), *Proceedings of ifip congress* (pp. 506–509). Macmillan and Co.

Tahiroğlu, K., Correia, N. N., & Espada, M. (2013, May). Pesi extended system: In space, on body, with 3 musicians. In *Proceedings of the international conference on new interfaces for musical expression* (pp. 35–40). Daejeon, Republic of Korea: Graduate School of Culture Technology, KAIST. Retrieved from http://nime.org/proceedings/2013/nime2013_97.pdf

Tahiroglu, K., Svedström, T., & Wikström, V. (2015, May). Musical engagement that is predicated on intentional activity of the performer with noisa instruments. In E. Berdahl & J. Allison (Eds.), *Proceedings of the international conference on new interfaces for musical expression* (pp. 132–135). Baton Rouge, Louisiana, USA: Louisiana State University. Retrieved from http://www.nime.org/proceedings/2015/nime2015_121.pdf

Tahiroğlu, K., Gurevich, M., & Knapp, R. B. (2018). Contextualising Idiomatic Gestures in Musical Interactions with NIMEs. In *Proceedings of the international conference on new interfaces for musical expression* (pp. 126–131). Virginia. Retrieved from http://www.nime.org/proceedings/2018/nime2018{_}paper0028.pdf

Tahiroğlu, K., Kildal, J., Ahmaniemi, T., Overstall, S., & Wikström, V. (2012). Embodied interactions with audio-tactile virtual objects in ahne. In *Proceedings of the 7th international conference on haptic and audio interaction design* (pp.

101–110). Berlin, Heidelberg: Springer-Verlag. Retrieved from `http://dx.doi.org/10.1007/978-3-642-32796-4_11` doi: 10.1007/978-3-642-32796-4_11

Tahiroğlu, K., Svedström, T., & Wikström, V. (2015). Noisa: A novel intelligent system facilitating smart interaction. In *Proceedings of the 33rd annual acm conference extended abstracts on human factors in computing systems* (pp. 279–282). New York, NY, USA: ACM. Retrieved from `http://doi.acm.org/10.1145/2702613.2725446` doi: 10.1145/2702613.2725446

Tahiroğlu, K., Vasquez, J. C., & Kildal, J. (2016). Non-intrusive counter-actions: Maintaining progressively engaging interactions for music performance. In *Proceedings of the international conference on new interfaces for musical expression.*

Välimäki, V., & Takala, T. (1996). Virtual musical instruments—natural sound using physical models. *Organised Sound*, *1*(2), 75–86.

Vasquez, J., Tahiroğlu, K., & Kildal, J. (2017). Idiomatic composition practices for new musical instruments: Context, background and current applications. In *Proceedings of the international conference on new interfaces for musical expression* (pp. 174–179). Copenhagen, Denmark: Aalborg University Copenhagen. Retrieved from `http://www.nime.org/proceedings/2017/nime2017_paper0033.pdf`

Vasquez, J. C., Tahiroğlu, K., & Kildal, J. (2017). Idiomatic composition practices for new musical instruments: Context, background and current applications. In *Proceedings of the international conference on new interfaces for musical expression.* Copenhagen, Denmark.

Waters, S. (2007). Performance ecosystems: Ecological approaches to musical interaction. *EMS: Electroacoustic Music Studies Network*, 1–20.

A Listings

(1) Custom version of the RoundedCube.cs script.

- `https://bitbucket.org/krrnk/thesis_listings/src/master/cs/RoundedCube.cs`

(2) Laser pointer and teleportation function.

- `https://bitbucket.org/krrnk/thesis_listings/src/master/cs/LaserPointer.cs`

(3) Modularity and routing in SuperCollider .

- `https://bitbucket.org/krrnk/thesis_listings/src/master/scd/modularity_routing.scd`

(4) Granular module used for the controllers.

- `https://bitbucket.org/krrnk/thesis_listings/src/master/scd/granularController.scd`

(5) Calculating velocity and acceleration in Unity 3d.

- `https://bitbucket.org/krrnk/thesis_listings/src/master/cs/acceleration.cs`

(6) Calculating average in SuperCollider.

- `https://bitbucket.org/krrnk/thesis_listings/src/master/scd/average.scd`

(7) FFT analysis DSP module.

- `https://bitbucket.org/krrnk/thesis_listings/src/master/scd/fftDSP.scd`

(8) Loudness analysis DSP module.

- `https://bitbucket.org/krrnk/thesis_listings/src/master/scd/loudness.scd`

(9) Amplitude analysis DSP module.

  - `https://bitbucket.org/krrnk/thesis_listings/src/master/scd/amplitudeTracker.scd`

(10) FFT analysis from SuperCollider to Unity.

  - `https://bitbucket.org/krrnk/thesis_listings/src/master/scd/fftToClient.scd`

(11) Applying force in Unity 3d.

  - `https://bitbucket.org/krrnk/thesis_listings/src/master/cs/force.cs`

(12) Getting Cartesian coordinates and Euler angles from head transformations.

  - `https://bitbucket.org/krrnk/thesis_listings/src/master/cs/head.cs`

(13) Left Controller's Cartesian coordinates and distance from a Raycast function.

  - `https://bitbucket.org/krrnk/thesis_listings/src/master/cs/raycast.cs`

(14) First implementation of the Ambisonic module.

  - `https://bitbucket.org/krrnk/thesis_listings/src/master/scd/atk1DSP.scd`

(15) User position and head rotation.

  - `https://bitbucket.org/krrnk/thesis_listings/src/master/scd/atk1Head.scd`

(16) Left Controller's sound source location in relation to the user.

  - `https://bitbucket.org/krrnk/thesis_listings/src/master/scd/atk1Controller.scd`

(17) Left Controller's Ambisonics and acceleration mapping.

  - `https://bitbucket.org/krrnk/thesis_listings/src/master/scd/atk1ControllerParams.scd`

(18) Freezing 3D structures in Unity.

- https://bitbucket.org/krrnk/thesis_listings/src/master/cs/freezing1.cs

(19) Freezing module DSP.

- https://bitbucket.org/krrnk/thesis_listings/src/master/scd/freezingDSP.scd

(20) Freezing module's control and Ambisonic module's amplitude mixing.

- https://bitbucket.org/krrnk/thesis_listings/src/master/scd/freezing1OSC.scd

(21) Pitch shifting module DSP.

- https://bitbucket.org/krrnk/thesis_listings/src/master/scd/shifterDSP.scd

(22) Trackpad values from Unity.

- https://bitbucket.org/krrnk/thesis_listings/src/master/cs/trackpadUnity.cs

(23) Trackpad mapping in SuperCollider.

- https://bitbucket.org/krrnk/thesis_listings/src/master/scd/trackpadSC.scd

(24) Environment activation according to the user's average acceleration.

- https://bitbucket.org/krrnk/thesis_listings/src/master/scd/environmentCounter.scd

(25) LFO mapping to spatial and rotation values of the Unity environment.

- https://bitbucket.org/krrnk/thesis_listings/src/master/scd/LFOmapping.scd

(26) Ambisonic and Granular modules mapping from environment's Raycast hit location.

- https://bitbucket.org/krrnk/thesis_listings/src/master/scd/environmentMapping.scd

(27) Polyhedrons' activation according to the user's average acceleration.

- `https://bitbucket.org/krrnk/thesis_listings/src/master/scd/polygonsCounter.scd`

(28) Polyhedrons' set's oscillations.

- `https://bitbucket.org/krrnk/thesis_listings/src/master/scd/polygonsOSC.scd`

(29) Colliders' categorization in Unity.

- `https://bitbucket.org/krrnk/thesis_listings/src/master/cs/controllerAllbutTrigger.cs`

(30) Colliders' categorization in SuperCollider.

- `https://bitbucket.org/krrnk/thesis_listings/src/master/scd/collider.scd`

(31) Ambisonic and Granular modules mapping polyhedron's position.

- `https://bitbucket.org/krrnk/thesis_listings/src/master/scd/polygonMapping.scd`

(32) Samples used in VIVR.

- `https://bitbucket.org/krrnk/thesis_listings/src/master/scd/samples.scd`

(33) Looper DSP module and OSC activation.

- `https://bitbucket.org/krrnk/thesis_listings/src/master/scd/looper.scd`

(34) Trigger pull down in Unity.

- `https://bitbucket.org/krrnk/thesis_listings/src/master/cs/trigger.cs`

(35) xFader DSP module and OSC activation.

- `https://bitbucket.org/krrnk/thesis_listings/src/master/scd/xFader.scd`

B Qualitative interview

1. How familiar are you with VR?
2. Have you tried any musical driven experience or musical instrument in VR?
3. How do you think VR can be used as a music tool?
4. What are your general thoughts after trying VIVR?
5. In VIVR the user is placed inside the instrument, do you think this influenced you when you were making music? If so, how?
6. Do you think VIVR has a targeted musical genre or does it offer enough musical possibilities to be used in diverse music genres?
7. How much the selected samples used in the Granular modules of VIVR influenced the music making?
8. Do you think that having the possibility to choose among different sets of samples on the fly would enhance the music experience?
9. Were you able to complete the task of creating a small composition based on the ABA' form?
10. Do you think it would be possible to create a whole track just using VIVR?

C Quantitative questionnaire

VIVR User test

This questionnaire is made to test the affordances of VIVR as a musical instrument. Your music skills will not be evaluated.

*Required

1. **Can you describe the sonic characteristics of the instrument? ***

2. **Have you used VR before? ***
   Mark only one oval.
   - Yes
   - No

3. **On a scale of 1 to 5, how familiar are you with VR? ***
   Mark only one oval.

	1	2	3	4	5	
I barely use it						I use it almost every day

4. **Have you experienced any musical environment/instrument in VR before? ***
   Mark only one oval.
   - Yes
   - No

5. **What was the name of it?**

6. **If you could describe VIVR with 3 words, what would they be? ***

7. **On a scale of 1 to 5, were the control mechanics easy to adapt to?** *
Mark only one oval.

	1	2	3	4	5	
Hard to adapt to	⬭	⬭	⬭	⬭	⬭	Easy to adapt to

8. **On a scale of 1 to 5, how intuitive were the control mechanics** *
Mark only one oval.

	1	2	3	4	5	
Unintuitive	⬭	⬭	⬭	⬭	⬭	Intuitive

9. **Did you perceive any latency between your interaction and the reaction response of the system?** *
Mark only one oval.

⬭ No
⬭ There's perceptible latency, but is not critical to the use of the instrument
⬭ The latency time makes the instrument mostly unusable.

10. **Did you feel any motion sickness during the experience?**
Mark only one oval.

⬭ Yes
⬭ No
⬭ Very little

11. **On a scale of 1 to 5, was the possibility to move sound in the 3 dimensions relevant as another musical element to play with?** *
Mark only one oval.

	1	2	3	4	5	
Irrelevant	⬭	⬭	⬭	⬭	⬭	Relevant, it was comparable to the use of dynamics or rhythm

12. **On a scale of 1 to 5, did the system's autonomous reactions add anything to the music you were making?** *
Mark only one oval.

	1	2	3	4	5	
No, they were disturbant	⬭	⬭	⬭	⬭	⬭	Yes, they felt supportive to the music I was making

13. **On a scale of 1 to 5, how uniform was the relationship between the sound output and the visual reactions?** *
Mark only one oval.

	1	2	3	4	5	
It was inconsistent. There was no relation between sound and visual output	◯	◯	◯	◯	◯	Consistent. Sound and visual output were very much connected

14. **On a scale of 1 to 5, how coherent was the mapping between movement interaction and audiovisual reactions?** *
Mark only one oval.

	1	2	3	4	5	
Unsystematic	◯	◯	◯	◯	◯	Coherent

15. **Did you ever get to feel that you were part of the instrument?** *
Mark only one oval.

◯ Yes
◯ No
◯ Maybe

16. **On a scale of 1 to 5, was the experience immersive?** *
Mark only one oval.

	1	2	3	4	5	
Uninteresting	◯	◯	◯	◯	◯	Immersive

17. **In the case that the experience was immersive, did this help you to get more engage with the music you were making?**
Mark only one oval.

◯ Yes
◯ No
◯ Maybe

18. **Did you notice the absence of haptic feedback?** *
Mark only one oval.

◯ Yes
◯ No